

**Universidade Federal de Ouro Preto  
Instituto de Ciências Exatas e Aplicadas  
Colegiado de Engenharia de Computação**



**UFOP**

Universidade Federal  
de Ouro Preto

**Simulador das práticas realizadas na  
disciplina de Maquinas Elétricas  
utilizando o motor de jogo Unity3D**

**Jhonson de Souza Rodrigues**

**TRABALHO DE  
CONCLUSÃO DE CURSO**

**ORIENTAÇÃO:**  
Thiago Luange Gomes

**Agosto, 2016  
João Monlevade/MG**

**Jhonson de Souza Rodrigues**

**Simulador das práticas realizadas na disciplina  
de Maquinas Elétricas utilizando o motor de  
jogo Unity3D**

Orientador: Thiago Luange Gomes

Coorientador: Alexandre Magno de Sousa

Monografia apresentada ao curso de Engenharia de  
Computação do Departamento de Computação e Sis-  
temas da Universidade Federal de Ouro Preto como  
requisito parcial para obtenção do grau de Bacharel  
em Engenharia de Computação

**Universidade Federal de Ouro Preto**

**João Monlevade**

**Agosto de 2016**



## Ata de Defesa

### ATA DE DEFESA

Aos 12 dias do mês de agosto de 2016, às 15 horas e 30 minutos, no Laboratório de Otimização e Expressão Gráfica(H102) do Instituto de Ciências Exatas e Aplicadas, foi realizada a defesa de Monografia pelo aluno Jhonson de Souza Rodrigues, sendo a Comissão Examinadora constituída pelos professores: Prof. Me. Thiago Luange Gomes, Prof. Me. Alexandre Magno de Sousa e Prof. Me. Tiago França Melo de Lima.

O candidato apresentou a monografia intitulada: "*Simulador das práticas realizadas na disciplina de Maquinas Elétricas utilizando o motor de jogo Unity3D*". A comissão examinadora deliberou, por unanimidade, pela aprovação do candidato, com nota 9.0 (Boa), concedendo-lhe o prazo de 15 dias para incorporação das alterações sugeridas ao texto final.

Na forma regulamentar, foi lavrada a presente ata que é assinada pelos membros da Comissão Examinadora e pelo graduando.

João Monlevade, 12 de agosto de 2016.

Prof. Me. Thiago Luange Gomes  
Professor Orientador/Presidente

Prof. Me. Alexandre Magno de Sousa  
Professor Coorientador

Prof. Me. Tiago França Melo de Lima  
Professor Convidado

Jhonson de Souza Rodrigues  
Graduando



UNIVERSIDADE FEDERAL DE OURO PRETO  
INSTITUTO DE CIÊNCIAS EXATAS E APLICADAS  
COLEGIADO DE ENGENHARIA DE COMPUTAÇÃO

---

## Folha de Aprovação

# Curso Engenharia de Computação

### FOLHA DE APROVAÇÃO DA BANCA EXAMINADORA

**Simulador das práticas realizadas na disciplina de  
Maquinas Elétricas utilizando o motor de jogo Unity3D**

**Jhonson de Souza Rodrigues**

**Monografia apresentada ao Departamento de Computação e  
Sistemas da Universidade Federal de Ouro Preto como  
requisito parcial da disciplina CSI496 – Trabalho de  
Conclusão de Curso II, do curso de Bacharelado em  
Engenharia de Computação e aprovada pela Banca  
Examinadora abaixo assinada:**

Prof. Me. Thiago Luange Gomes

Departamento de Computação e Sistemas/DECSI - UFOP

Prof. Me. Alexandre Magno de Sousa

Departamento de Computação e Sistemas/DECSI - UFOP

Prof. Me. Tiago França Melo de Lima

Departamento de Computação e Sistemas/DECSI - UFOP

João Monlevade, 12 de agosto de 2016



UFOP  
Universidade Federal  
de Ouro Preto

UNIVERSIDADE FEDERAL DE OURO PRETO  
INSTITUTO DE CIÊNCIAS EXATAS E APLICADAS  
COLEGIADO DE ENGENHARIA DE COMPUTAÇÃO

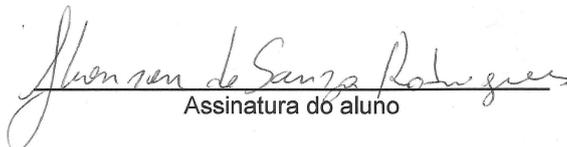
---

**TERMO DE RESPONSABILIDADE**  
**Curso Engenharia de Computação**

**TERMO DE RESPONSABILIDADE**

Eu, **Jhonson de Souza Rodrigues**, declaro que o texto do trabalho de conclusão de curso intitulado **“Simulador das práticas realizadas na disciplina de Maquinas Elétricas utilizando o motor de jogo Unity3D”** é de minha inteira responsabilidade e que não há utilização de texto, material fotográfico, código fonte de programa ou qualquer outro material pertencente a terceiros sem as devidas referências ou consentimento dos respectivos autores.

João Monlevade, 12 de agosto de 2016

  
Assinatura do aluno

R696s

Rodrigues, Jhonson de Souza.

Simulador das práticas realizadas na disciplina de máquinas elétricas utilizando o motor de jogo Unity3D [manuscrito] / Jhonson de Souza Rodrigues. - 2017.

68f.: il.: color; tabs.

Orientador: Prof. Me. Thiago Luange Gomes.

Coorientador: Prof. Me. Alexandre Magno de Sousa.

Monografia (Graduação). Universidade Federal de Ouro Preto. Instituto de Ciências Exatas e Aplicadas. Departamento de Computação e Sistemas de Informação.

1. Simulação (Computadores). 2. Máquinas elétricas. 3. Computação gráfica. 4. Jogos eletrônicos. I. Gomes, Thiago Luange. II. Sousa, Alexandre Magno de. III. Universidade Federal de Ouro Preto. IV. Título.

Catálogo: [ficha@sisbin.ufop.br](mailto:ficha@sisbin.ufop.br)

CDU: 004.9

# Agradecimentos

Primeiramente a Deus que permitiu que tudo acontecesse. Aos meus pais, Francisco e Gilda, e à minha irmã, Franceila, pelo amor, incentivo e apoio incondicional. Aos meus orientadores, Thiago e Alexandre, pelo empenho dedicado à elaboração deste trabalho. A minha namorada, Larissa, pelo companheirismo, amor e apoio na correria de cada semestre. E a todos que direta ou indiretamente fizeram parte da minha formação, o meu muito obrigado.

*“Pai querido obrigado pelos planos.  
Grande sonhos, especiais pra mim.  
Tu me guias nos caminhos desta vida  
Ao teu lado nada faltara...  
...Quando fraco, tua força nunca falhará“  
(Canção da Vida, Lineu Soares)*

# Resumo

A construção de um simulador para atividades práticas de diferentes disciplinas é considerada uma abordagem pedagógica interessante, já que ele permite aos discentes vivenciar experiências próximas à realidade, deixando-os motivados na construção do conhecimento, o que auxilia o processo de ensino e aprendizagem. É seguindo esse percurso que este trabalho propõe o desenvolvimento de um simulador para as atividades práticas realizadas na disciplina Máquinas Elétricas, ofertada no curso de Engenharia Elétrica. Ao longo deste trabalho são abordados os principais passos e ferramentas para o desenvolvimento de um simulador: modelagem tridimensional, montagem de cena e técnicas de computação gráfica. Para tanto, foram obtidos componentes com características visuais próximas às encontradas em laboratório. Os valores obtidos nos testes de simulação são próximos aos encontrados em laboratórios. Algumas dificuldades foram enfrentadas. Porém, pode-se destacar a representação dos cabos, devido deformação não linear no espaço, os cálculos da simulação e a quantidade de possibilidades de interação com os usuários, como as dificuldades mais recorrentes. Como contribuição para trabalhos futuros, novas práticas poderão ser inseridas a este trabalho: melhorias nos polígonos e texturas objetos modelos, além de ajustes na interação com os usuários são apenas alguns exemplos.

**Palavras-chaves:** simulador, máquinas elétricas, computação gráfica, jogos eletrônicos, games, modelagem 3D.

# Abstract

The construction of a simulator to practice of different disciplines activities is seen as an interesting pedagogical approach, since it helps the cognitive learning process. This work proposes the development of a simulator for the practical activities carried out in the Electrical Machines discipline, offered to Electrical Engineering course. We consider throughout this work the major steps and tools for the development of a simulator: a three-dimensional modeling, scene assembly and computer graphics techniques. The values obtained in this search process are close to those found in laboratories. Some difficulties were faced, especially the representation of the cables, the calculations of simulation and interaction with users. As a contribution to future search new practices may be inserted to this work, in addition to improvements in the models applied.

**Key-words:** simulator, electric machines, Computer graphics, electronic game, 3D modeling.

# Lista de ilustrações

Figura 1 – Principais tipos de fonte de luz adaptada de (COHEN, 2006). . . . .	21
Figura 2 – Componentes da iluminação adaptada de (COHEN, 2006). . . . .	21
Figura 3 – As propriedades de refração da luz em <i>Bump Map</i> (AZEVEDO, 2003). . . . .	22
Figura 4 – Paredes na esquerda sem <i>Bump Map</i> e da direita com <i>Bump Map</i> (AZEVEDO, 2003). . . . .	23
Figura 5 – Diagrama simplificado de uma máquina editado a partir de Chapman (CHAPMAN, 2013). . . . .	26
Figura 6 – Circuito equivalente de um motor de corrente contínua em derivação (CHAPMAN, 2013). . . . .	27
Figura 7 – Interface da área de trabalho do Blender. . . . .	33
Figura 8 – Interface da área de trabalho do Unity3D. . . . .	33
Figura 9 – Circuito de conexão do motor corrente contínua do tipo shunt. . . . .	35
Figura 10 – Protótipo da cena com as posições dos objetos marcados por caracteres. . . . .	37
Figura 11 – Mapa com os números dos bornes. . . . .	38
Figura 12 – Tipos de associações dos resistores. . . . .	38
Figura 13 – Desenho da bancada didática. . . . .	39
Figura 14 – Adicionando uma imagem no <i>background</i> . . . . .	39
Figura 15 – Resultado da modelagem da vista lateral direita. . . . .	40
Figura 16 – Resultado da modelagem da vista frontal. . . . .	41
Figura 17 – Bancada didática logo após a extrusão da face lateral. . . . .	42
Figura 18 – Resultado da Unwrap do objeto da bancada didática. . . . .	42
Figura 19 – Exportando o mapa de textura. . . . .	43
Figura 20 – <i>UVs Map</i> da bancada didática. . . . .	44
Figura 21 – Modelo da bancada didática após aplicação de textura. . . . .	44
Figura 22 – Curva de Bézier no Blender, os pontos de controle da curvas são representados pelo cor amarela, os pontos de comprimento são representados pelo cor vermelha. . . . .	45
Figura 23 – Menu com os <i>modifiers</i> . . . . .	46
Figura 24 – Atribuindo uma curva <i>Bézier</i> ao cilindro. . . . .	46
Figura 25 – Resultado da modelagem dos cabos juntos com os demais objetos. . . . .	47
Figura 26 – Painéis da área de trabalho Unity3D. . . . .	47
Figura 27 – Algumas opções de <i>collider</i> no Unity. . . . .	49
Figura 28 – Painel do <i>Inspector</i> com as informações do variador de restado. . . . .	50
Figura 29 – Interface do menu inicial. . . . .	51
Figura 30 – Interface da cena principal. . . . .	52
Figura 31 – Janelas de ajuda ao usuário. . . . .	53

Figura 32 – Objetos com variador . . . . .	53
Figura 33 – Realce de um conector devido a sua seleção. . . . .	54
Figura 34 – Realce de um cabo devido a sua seleção. . . . .	55
Figura 35 – Simulação com a fonte em série e o voltímetro. . . . .	55
Figura 36 – Simulação com a fonte em série, voltímetro, miliamperímetro, reostato e o enrolamento de campo. . . . .	56
Figura 37 – Simulação da conexão em paralelo da carga resistiva. . . . .	57
Figura 38 – Simulação com todos os equipamentos conectados. . . . .	57
Figura 39 – Simulador em funcionamento . . . . .	58
Figura 40 – Objeto 3D referente a bancada didática . . . . .	65
Figura 41 – Objeto 3D referente ao motor . . . . .	65
Figura 42 – Objeto 3D referente ao reostato . . . . .	66
Figura 43 – Objeto 3D referente da carga resistiva . . . . .	66
Figura 44 – Foto da Bancada Didática . . . . .	67
Figura 45 – Foto do Reostato . . . . .	67
Figura 46 – Foto do Motor . . . . .	68
Figura 47 – Foto da Carga Resistiva . . . . .	68

# Lista de tabelas

Tabela 1 – Requisitos mínimos para utilização das ferramentas e sistema utilizado.	34
Tabela 2 – Tabela com os pares de conexões possíveis para a prática. . . . .	37
Tabela 3 – Velocidades do motor pela resistência da armadura . . . . .	57
Tabela 4 – Comparação entre os valores obtidos pelo simulador com os valores obtidos no laboratórios . . . . .	58

# Lista de abreviaturas e siglas

fbx	<i>Filmbox</i>
RV	Realidade Virtual
CAD	<i>Computer Aided Design</i>
IBM	<i>International Business Machines</i>
UI	<i>User Interfaces</i>
rpm	Rotações por minuto
XP	<i>Extreme Programming</i>
RUP	<i>Rational Unified Process</i>
3D	Tridimensional
API	<i>Application Programming Interface</i>
NaN	<i>Not a Number</i>
MS	Microsoft
GPL	<i>General Public License</i>
IDE	<i>Integrated Development Environment</i>
DC	<i>Direct current</i>
V	Unidade de medida para a tensão elétrica
A	Unidade de medida para intensidade de corrente elétrica
$\Omega$	Unidade de medida para resistência elétrica

# Sumário

<b>1</b>	<b>INTRODUÇÃO</b>	<b>14</b>
<b>2</b>	<b>FUNDAMENTAÇÃO TEÓRICA</b>	<b>17</b>
<b>2.1</b>	<b>Processo de Desenvolvimento de um Simulador</b>	<b>17</b>
2.1.1	Ciclo de vida do projeto	18
2.1.2	Profissionais envolvidos	18
<b>2.2</b>	<b>Computação Gráfica</b>	<b>19</b>
2.2.1	Câmera virtual	20
2.2.2	Iluminação	20
2.2.3	Textura	22
<b>2.3</b>	<b>Modelagem 3D</b>	<b>23</b>
2.3.1	Softwares de Modelagem 3D	24
<b>2.4</b>	<b>Motores de Jogo</b>	<b>24</b>
<b>2.5</b>	<b>Máquinas Elétricas</b>	<b>25</b>
<b>2.6</b>	<b>Considerações Finais</b>	<b>28</b>
<b>3</b>	<b>PESQUISAS RELACIONADAS</b>	<b>29</b>
<b>3.1</b>	<b>Ambiente multimídia para o ensino de máquinas elétricas: Uma Abordagem Multidisciplinar</b>	<b>29</b>
<b>3.2</b>	<b>Estudo de máquinas elétricas auxiliado por técnicas de realidade virtual</b>	<b>30</b>
<b>3.3</b>	<b>Um Laboratório virtual tridimensional e interativo para suporte ao ensino de física</b>	<b>30</b>
<b>3.4</b>	<b>Considerações Finais</b>	<b>31</b>
<b>4</b>	<b>METODOLOGIA E DESENVOLVIMENTO</b>	<b>32</b>
<b>4.1</b>	<b>Recursos Computacionais e Ferramentas</b>	<b>32</b>
<b>4.2</b>	<b>Prática do controle de velocidade de um motor Shunt</b>	<b>34</b>
<b>4.3</b>	<b>Desenvolvimento</b>	<b>35</b>
4.3.1	Especificação do Sistema	35
4.3.2	Modelagem dos Objetos	38
4.3.3	Construção da Cena	47
4.3.4	<i>Scripts</i>	49
<b>4.4</b>	<b>Considerações Finais</b>	<b>50</b>
<b>5</b>	<b>RESULTADOS</b>	<b>51</b>

<b>5.1</b>	<b>Interface . . . . .</b>	<b>51</b>
<b>5.2</b>	<b>Funcionamento . . . . .</b>	<b>53</b>
<b>5.3</b>	<b>Considerações Finais . . . . .</b>	<b>59</b>
<b>6</b>	<b>CONCLUSÃO . . . . .</b>	<b>60</b>
	<b>REFERÊNCIAS . . . . .</b>	<b>61</b>
	<b>APÊNDICES</b>	<b>64</b>

# 1 Introdução

A indústria de jogos, que abrange os jogos eletrônicos, simuladores e realidade virtual, detém o maior faturamento entre as indústrias de entretenimento, obtendo ganhos de U\$ 91,8 Bilhões em 2015, com estimativa de crescimento de 8,5U% para o ano de 2016. A projeção entre os anos de 2015 a 2019 é de 29,5% alcançando um faturamento de U\$ 118,6 Bilhões em 2019 (NEWZOO, 2016). A indústria cinematográfica, segunda colocada, obteve um faturamento de U\$ 11,5 Bilhões em 2015 (MOJO, 2016), aproximadamente 800% menor que o da indústria de jogos. Essa comparação ajuda a compreender melhor o papel protagonista da indústria de jogos na indústria de entretenimento. Os ganhos alcançados pela indústria de jogos faz expandir as pesquisas e a relevância de simuladores, levando-os, em diferentes países, a ser considerado um dos setores mais estratégicos da indústria tecnológica. Além disso, suas ferramentas interativas de visualização e simulação podem ser utilizadas em diversas áreas como: engenharia (estruturas, automobilística, aeroespacial), exploração de petróleo, medicina, treinamento de pilotos, entre outras aplicações (FEIJO B., 2006).

O treinamento por meio de simuladores e ferramentas interativas de visualização é muito importante em diversas áreas do conhecimento e profissional. Isso porque para capacitar os profissionais que irão exercer determinada tarefa é necessário, em muitos casos, colocar o profissional em situação real de trabalho, manuseando máquinas e equipamentos. Esse treinamento tem um custo muito elevado, já que o equipamento fica parado para o treinamento, quando poderia ser utilizado na produção. Uma maneira de evitar essa situação onerosa é iniciar o treinamento em um simulador e, por fim, finalizar em máquinas e equipamentos reais (LIMONGELLI et al., 2012). Na simulação é possível reproduzir todos os elementos encontrados em máquinas e equipamentos, inclusive eventos críticos e perigosos, que muitas vezes não são possíveis em treinamentos convencionais. Além disso, a simulação diminui a chance de danificar equipamentos durante um treinamento (ROCHA et al., 2013).

Para além do caráter de treinamento, a simulação também pode ser usada na educação, tanto para o uso de práticas laboratoriais, como em um pátio de produção, em hidroelétricas, na simulação de uma cirurgia, trazendo a realidade mais próxima ao aprendizado do aluno (JUNIOR; CARDOSO; JR, 2008). A simulação também pode ser utilizada para exemplificação de conteúdos disciplinares, já que os alunos assimilam melhor o conteúdo apresentado em formato multimídia e por meio de interações (BARILLI; EBECKEN; CUNHA, 2011).

No que se refere ao ensino, sobretudo em atividades práticas, os simuladores são

utilizados de diferentes maneiras. Eles atuam como ferramenta de suporte no processo cognitivo, adequando-se a diferentes formas de aprendizagem (BARILLI; EBECKEN; CUNHA, 2011), o que possibilita a produção de ambientes educacionais à distância com um custo acessível e contribui para qualidade desses processos (JUNIOR; CARDOSO; JR, 2008). Outra justificativa para o uso de ambientes virtuais no contexto de aprendizagem é o desafio de ensinar atividades práticas de forma segura, pois há atividades práticas que envolvem elementos que podem apresentar risco aos alunos (LIMONGELLI et al., 2012).

Diversas instituições de ensino não possuem equipamentos laboratoriais adequados, ou seja, com uma infraestrutura que possibilite a abordagem de aspectos importantes relacionados às disciplinas. A falta de equipamentos e o mau funcionamento são os principais problemas. Um dos fatores para a insuficiência de equipamentos laboratoriais na maior parte das instituições de ensino brasileira, talvez seja seu custo elevado. Esse fator faz com que se demande grande atenção de educadores e estudantes a fim de serem evitados acidentes que possam causar danos às pessoas ou aos equipamentos. As montagens incorretas, por exemplo, é uma das mais frequentes ocorrências causadoras de danos aos equipamentos (ROCHA et al., 2013).

Um exemplo claro do cenário descrito anteriormente pode ser visto na disciplina de Máquinas Elétricas, ministrada no curso de Engenharia Elétrica. Essa disciplina exige o exercício prático em equipamentos de alto custo (JUNIOR; CARDOSO; JR, 2008). Devido ao custo elevado, diversos laboratórios de máquinas elétricas, em diferentes instituições de ensino, não possuem uma infraestrutura que permita a abordagem de aspectos importantes para o processo de ensino e aprendizagem dessa disciplina (ROCHA et al., 2013).

Diante deste cenário, este trabalho propõe desenvolver um simulador de atividades práticas e avaliar os benefícios deste sistema para os estudantes. É importante notar que o simulador de atividades práticas tem como referencial a prática de controle de velocidade de um motor de corrente contínua. A escolha por esse objeto de estudo se justifica em função do perigo apresentado durante a execução desse exercício prático já que, além de alta tensão, são utilizados vários componentes presentes no laboratório. Neste trabalho são apresentados alguns conceitos, assim como as etapas para o desenvolvimento de um simulador e os detalhes acerca de seu funcionamento.

O simulador de práticas de Máquinas Elétricas apresentado aqui é um ambiente virtual que possibilita ao usuário interagir com os objetos e, assim, realizar a montagem correta da atividade prática de controle de velocidade de um motor de corrente contínua. Permite ainda, a instrução do aluno sobre tentativas de ligações ou montagens incorretas.

Ao final, foram obtidos componentes com características visuais próximas às encontradas em laboratório, aumentando a familiarização dos alunos com as máquinas e os equipamentos. Também foi possível realizar o controle da velocidade de rotação do motor de três diferentes modos: variação de tensão, corrente de campo e da resistência

da armadura. Por fim, é importante salientar que o ambiente virtual permite ao aluno comparar os valores da velocidade do motor, da tensão aplicada, da corrente de campo e da corrente da armadura com os valores encontrados em laboratório real.

A fundamentação teórica deste trabalho é desenvolvida no segundo capítulo. Na ocasião são abordadas as metodologias aplicadas para o desenvolvimento de um simulador, bem como a formulação teórica da computação gráfica: conceitos de modelagem tridimensional, motores de jogos e de máquinas elétricas. No terceiro capítulo são descritos alguns trabalhos afins a esta pesquisa. No quarto capítulo a metodologia empregada e o desenvolvimento são apresentados. No quinto capítulo são apresentados os resultados obtidos, ou seja, as interfaces de seu funcionamento. Por fim, no sexto capítulo, a conclusão final desta pesquisa é estruturada.

## 2 Fundamentação Teórica

A simulação é uma área do conhecimento que trabalha de maneira interdisciplinar. Portanto, o desenvolvimento de uma pesquisa a respeito de suas funções exige a introdução de conceitos básicos de diferentes áreas do conhecimento. Este trabalho não foge à regra.

Pode-se definir simulação como o processo de implementação de modelos que permitem a visualização e a manipulação de objetos em ambientes tridimensionais criados por computadores e transmitidos aos usuários através de periféricos (FEIJO B., 2006). São ambientes virtuais interativos, cujas possibilidades são limitadas pela imaginação do desenvolvedor (TORI CLAUDIO KIRNER, 2006). Os principais conceitos envolvidos no desenvolvimento de simuladores, tendo como foco a caracterização de assuntos diretamente ligados a este trabalho, são apresentados a seguir.

Este capítulo divide-se em seis seções. A seção 2.1 aborda os procedimentos necessários para o desenvolvimento de um simulador, metodologias, o ciclo de vida do projeto e os profissionais envolvidos. A seção 2.2 apresenta alguns conceitos relacionados à computação gráfica. A seção 2.3 descreve a modelagem 3D e alguns softwares. Na seção 2.4 são descritos os motores de jogo. A seção 2.5 apresenta os conceitos básicos relacionados a máquinas elétricas de corrente contínua. E por fim, a seção 2.6 apresenta as considerações finais deste capítulo.

### 2.1 Processo de Desenvolvimento de um Simulador

Existem várias etapas para a concepção de um simulador ou qualquer outro tipo de software (ARAUJO, 2006). No entanto, o desenvolvimento de um simulador tem como característica principal a multidisciplinaridade, aspecto que o diferencia dos demais softwares. Não são apenas programadores que trabalham na estruturação de um simulador. Profissionais de áreas distintas como desenhistas, músicos, escritores, roteiristas, especialistas em banco de dados e em redes, profissionais que trabalham com inteligência artificial atuam no desenvolvimento de simuladores.

Os trabalhos de Barros e Araújo (BARROS, 2007; ARAUJO, 2006) apresentam as principais metodologias aplicadas no desenvolvimento de um jogo ou simulador. Eles destacam as seguintes metodologias: *Game Waterfall Process*, *Extreme Game Development*, *Scrum*, *Game Unified Process*. Com exceção da metodologia *Scrum*, aplicada em sua íntegra, as demais metodologias são adaptações para o contexto de jogos de metodologias já existentes. Sendo que a metodologia *Game Waterfall Process* é baseada na tradicional metodologia em cascata, a metodologia *Extreme Game Development* tem como base a

metodologia *Extreme Programming* (XP), e a metodologia *Game Unified Process* uni a metodologia XP juntamente com RUP (*Rational Unified Process*). Esta última é uma metodologia de propriedade da IBM.

### 2.1.1 Ciclo de vida do projeto

De acordo com Sloper (SLOPER, 2002) o ciclo de vida de um projeto de jogos ou simuladores é constituído de cinco etapas: concepção, pré-produção, produção, pós-produção e pós-lançamento. Elas são bem definidas e correlacionadas. Vejamos uma melhor definição dessas etapas:

- **Concepção:** neste momento são realizados estudos para atestar a viabilidade do projeto. Posteriormente é apresentado um documento conceitual (BARROS, 2007).
- **Pré-produção:** nesta etapa é gerado um documento conhecido como *Game Design*. Ele contém os elementos necessários para o desenvolvimento do jogo, tais como: o enredo, as dinâmicas de jogo e os objetivos (ARAUJO, 2006).
- **Produção:** durante esta fase o documento de arte é confeccionado. Aqui, no estágio de desenvolvimento do projeto de jogos, a equipe de arte produz os pormenores dos cenários, as personagens e os objetos. É neste momento que as músicas são escolhidas, os efeitos sonoros criados, e o documento de som é engendrado. Nesta fase também ocorre o desenvolvimento de códigos fontes, a modelagem dos objetos e a criação de cenários, tudo baseado no *Game Design*. Ao final desta etapa, várias versões de teste são criadas até a criação do produto final (BARROS, 2007).
- **Pós-Produção:** a equipe de teste verifica possíveis *bugs* e os reporta à equipe de desenvolvimento. Nesta etapa, a equipe de marketing elabora estratégias publicitárias e a campanha de lançamento (ARAUJO, 2006).
- **Pós-lançamento:** após o lançamento, o comportamento e as reações do público sobre o jogo são estudadas a fim de oferecer um suporte a varejistas e consumidores. Neste momento, a criação de novas versões e atualizações, bem como, possíveis expansões na estrutura dos jogos são avaliadas (BARROS, 2007).

### 2.1.2 Profissionais envolvidos

Os profissionais envolvidos na construção de um jogo ou de um simulador são variáveis de acordo com o projeto. No entanto, alguns profissionais são extremamente importantes (ARAUJO, 2006). Conforme apresentado por Gamasutra (GAMASUTRA, 2014), os principais profissionais envolvidos na produção de um jogo ou simulador são:

- Produtor: ele é o responsável direto pelo projeto. Em um software tradicional os requisitos necessários para o desenvolvimento do projeto são levantados por meio de interação com o cliente. No desenvolvimento de jogos ou de simulador, porém, o produtor é o responsável por enumerar os requisitos e, também, fiscalizar cada momento da confecção do jogo ou do simulador (SLOPER, 2002).
- *Game Designer*: é responsável pela definição dos aspectos do jogo, sua jogabilidade, o comportamento de personagens e objetos, as características sonoras e visuais do jogo, a concepção dos cenários e a descrição dos objetivos. Esse profissional pode, também, em alguns momentos ser responsável pela construção dramática do jogo, substituindo as funções do roteirista (ARAUJO, 2006).
- Artista: eles estão diretamente ligados à concepção dos aspectos e elementos visuais e artísticos dos jogos. Ele também exerce um papel importante na criação de objetos e personagens, na modelagem 3D, animação e criação de texturas (CARVALHO, 2006).
- Programador: é o responsável pela criação de funções, módulos, entidades e componentes importantes para a interação e o funcionamento do jogo (ARAUJO, 2006).
- Músicos: esses profissionais estão ligados às características sonoras do jogo. Trabalham na concepção dos efeitos e da trilha sonora. Podem ser responsáveis por essa etapa do desenvolvimento do jogo tanto um único músico quanto uma orquestra (BARROS, 2007).
- *Quality Testers*: responsável por encontrar e reportar possíveis *bugs* e erros no jogo. Ele é diretamente responsável pela qualidade do jogo produzido. Esse profissional deve ficar atento aos níveis de qualidade exigidos pelo mercado, realizar testes de aceitação pelos usuários e testes de usabilidade e jogabilidade (SLOPER, 2002).

## 2.2 Computação Gráfica

A computação gráfica é uma ferramenta computacional que, através da matemática, proporciona a criação de imagens (COHEN, 2006). Com ela é possível transcender técnicas artísticas tradicionais e criar mundos virtuais, bem como, desenvolver representações da realidade com uma riqueza de detalhes muitas vezes não alcançada em ferramentas artísticas tradicionais possibilitando, assim, novos horizontes. A computação gráfica é a junção perfeita entre matemática, arte e computação (AZEVEDO, 2003). Na intenção de desenvolver mundos virtuais mais próximos à realidade, a computação gráfica precisa abordar as principais características relacionadas à formação de imagens, tais como câmera, estrutura da cena, iluminação e textura (HEARN M. PAULINE BAKER, 2011).

### 2.2.1 Câmera virtual

Uma fotografia produzida em uma máquina fotográfica real é criada através da projeção da cena em um plano. Partindo desse conceito uma câmera virtual pode trabalhar com a simulação de uma câmera real que projeta a cena 3D virtual em um plano para criar uma imagem. Podemos entender a câmera virtual como sendo os olhos de um observador (COHEN, 2006).

Para a definição completa de uma câmera virtual são necessários dois tipos básicos de parâmetros: parâmetros relacionados ao posicionamento da câmera no espaço posição da câmera, direção para onde a câmera está sendo apontada- e o *view up* o vetor que indica o lado acima da câmera. Há ainda um segundo grupo de parâmetros que podem ser entendidos como parâmetros diretamente relacionados à câmera, tais como: ponto focal, ângulo de abertura e tamanho de janela de visualização (AZEVEDO, 2003) (HEARN M. PAULINE BAKER, 2011).

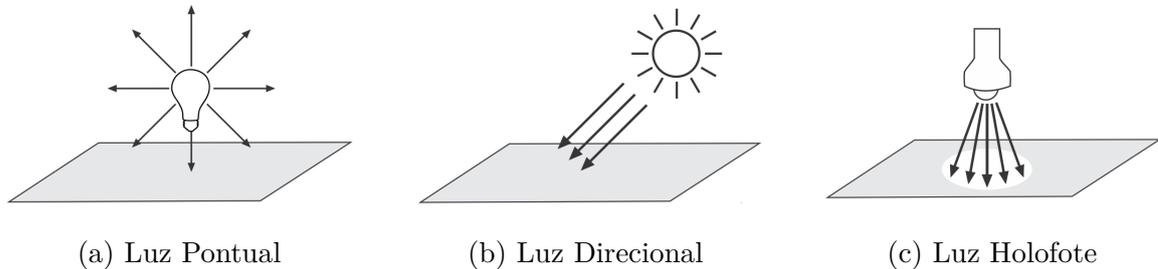
### 2.2.2 Iluminação

A iluminação da cena, juntamente com as propriedades do material, é responsável por definir a cor do objeto. Essa etapa é extremamente importante para o realismo da imagem gerada. Desse modo, é necessário descrever as fontes de luz e a relação delas com os materiais. Segundo Azevedo (AZEVEDO, 2003) pode-se simplificar a descrição das fontes de luz com três tipos básicos: luz direcional, luz pontual e luz holofote (Figura 1) (COHEN, 2006):

- Luz Pontual: a partir de um único ponto todos os raios são emitidos de maneira uniforme em todas as direções. A intensidade varia de acordo com a distância do ponto da luz até o objeto (Figura 1a) (PHARR; HUMPHREYS, 2010).
- Luz Direcional: esse tipo de iluminação possui uma direção e intensidade única, sendo muito utilizada para simular a luz solar. Sua cor, posição e ângulo podem ser ajustados (Figura 1b) (COHEN, 2006).
- Luz Holofote: a luz holofote é emitida tendo direção e sentido previamente definidos, podendo ocorrer variações de intensidade no que tange a distância do objeto. Seu diferencial está em ser uma luz focada, bem semelhante a uma lanterna. Intensidade, posição e ângulo podem ser ajustados (Figura 1c) (AZEVEDO, 2003) (PHARR; HUMPHREYS, 2010).

Faz-se necessário agora, após as análises das fontes de luz, descrever como a luz proveniente de uma fonte interage com a superfície de um material, para em seguida produzir a cor da cena (COHEN, 2006). Baseando-se nas propriedades da superfície

descritas no modelo de iluminação de Phong, é possível caracterizar a interação da superfície com a iluminação a partir de três componentes: ambiente, difusa e especular 2) (HEARN M. PAULINE BAKER, 2011).



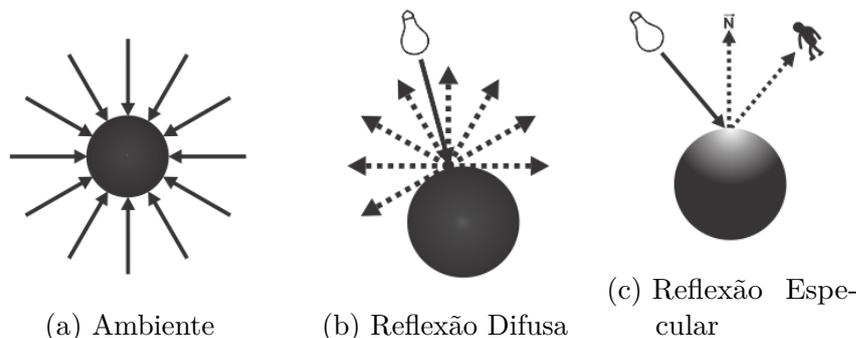
(a) Luz Pontual

(b) Luz Direcional

(c) Luz Holofote

Figura 1 – Principais tipos de fonte de luz adaptada de (COHEN, 2006).

- Ambiente: na luz ambiente não há uma direção definida. Ela incide com igual intensidade em todos os objetos da cena, simulando a reflexão vinda de todas as superfícies difusas. Ela é muito utilizada em simulação de ambientes externos, ajudando a melhorar o realismo dramático. Ela também determina o nível mínimo de iluminação das superfícies (Figura 2a) (AZEVEDO, 2003).
- Reflexão Difusa: reflexão difusa ou lambertiana pode radiar a luz incidente em todas as direções. A luz refletida e percebida será a mesma independente da posição do observador. Esse tipo de reflexão da luz pode ser trabalhado em superfícies foscas. Uma das características da reflexão difusa é a absorção da luz pela superfície. Esta, por sua vez, reflete a cor (Figura 2b) (COHEN, 2006);
- Reflexão Especular: são reflexões provenientes de superfícies brilhosas, polidas, onde a luz incidida é refletida diretamente no observador. Na reflexão especular a percepção da reflexão depende da posição da luz e do observador. É importante notar que, neste modelo, a luz refletida será a mesma que incide na superfície (Figura 2c) (PHARR; HUMPHREYS, 2010).



(a) Ambiente

(b) Reflexão Difusa

(c) Reflexão Especular

Figura 2 – Componentes da iluminação adaptada de (COHEN, 2006).

### 2.2.3 Textura

Na computação gráfica a textura é fundamental para a representação realista e para o alcance da representação gráfica do objeto desejada (COHEN, 2006). Uma técnica de aplicação de textura muito interessante, conhecida como *texture map*, possibilita a aplicação de imagens em superfícies específicas do objeto. Essa técnica consiste na aplicação de uma textura para que, logo em seguida, o pixel da superfície tenha o valor pixel correspondente à textura ou à combinação dele com a cor proveniente do cálculo da iluminação. Seu baixo custo de processamento torna-o um procedimento muito interessante (AZEVEDO, 2003).

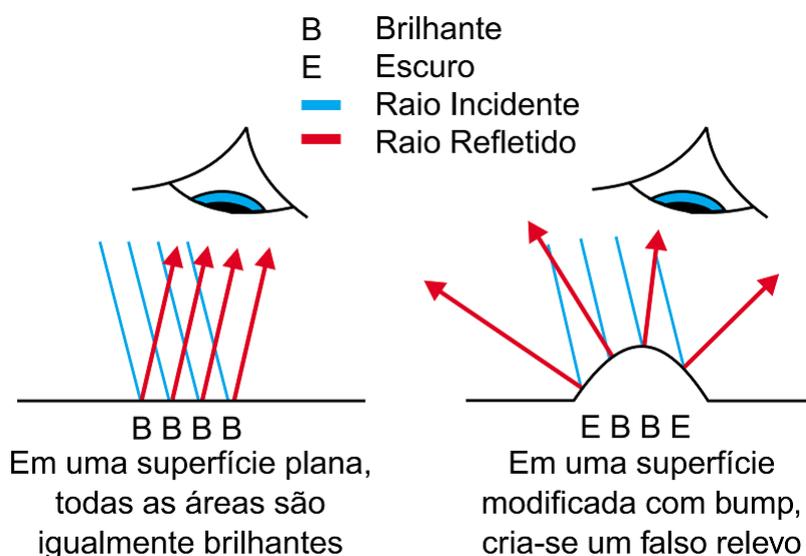


Figura 3 – As propriedades de refração da luz em *Bump Map* (AZEVEDO, 2003).

Há diversas outras técnicas baseadas em texturas cuja função é proporcionar uma representação mais realista dos objetos. Uma delas é conhecida como *Bump Map*. Essa técnica tem como uma de suas principais características a adição de uma perturbação no vetor normal (vetor perpendicular ao plano da superfície) de cada pixel que representa um relevo no objeto. Como a cor da superfície está relacionada ao seu vetor normal e à direção da luz, a perturbação através das propriedades de refração da superfície acarretará em partes mais claras ou mais escuras no objeto (AZEVEDO, 2003). Na Figura 3, situada logo acima, pode-se observar alterações provenientes dessa técnica por meio da qual os raios incidentes mudam de direção de acordo com as partes mais brilhantes ou escuras do *Bump Map*. Na Figura 4, que pode ser observada logo abaixo, duas paredes são ordenadas lado a lado: a da esquerda, sem *Bump Map*, e da direita, com a aplicação do *Bump Map*. Na parede à direita, com *Bump Map*, as imperfeições do tijolo, bem como o rebaixamento entre eles é bem visível, o que torna a representação mais realista. Essas alterações na textura são resultados da aplicação do *Bump Map*.

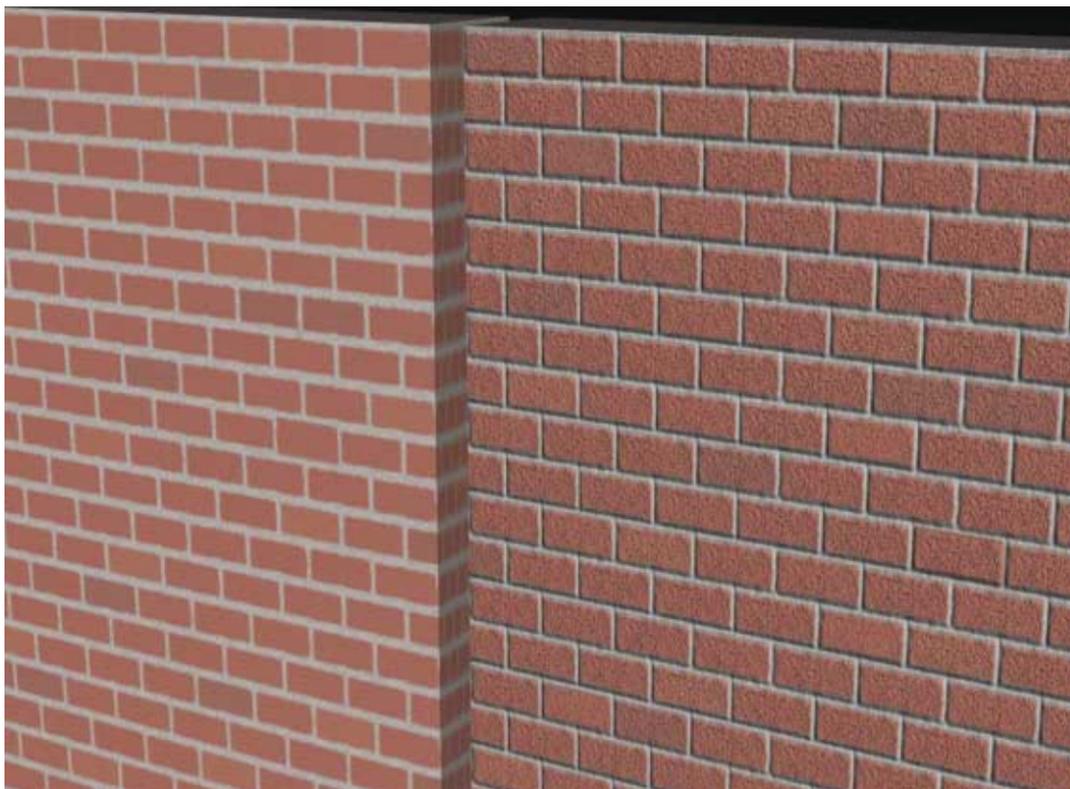


Figura 4 – Paredes na esquerda sem *Bump Map* e da direita com *Bump Map* (AZEVEDO, 2003).

## 2.3 Modelagem 3D

Como já dito, estão disponíveis no mercado diversas técnicas para a modelagem de objetos. Algumas delas são eficientes para determinados tipos de objetos e ineficientes para outros. Desse modo, cada técnica deve ser adotada de acordo com o problema (AZEVEDO, 2003). As técnicas de modelagem podem ser classificadas em dois grupos: modelagem de não sólidos- que não apresentam forma definida como, por exemplo, água, fumaça, tecido e etc (HEARN M. PAULINE BAKER, 2011) e a modelagem de sólidos. Esta última consiste em utilizar técnicas que permitam a criação de objetos através de suas propriedades geométricas (esferas, cubos, cilindros, pirâmides e dentre outros) ou através de adição intersecção, subtração e corte desses mesmos sólidos (AZEVEDO, 2003).

Pode-se dizer que o define computacionalmente a estrutura de objetos são os pontos. Esses são chamados de vértices e definidos por uma terna  $(x,y,z)$ , sendo o elemento geométrico mais básico (CLUA; BITTENCOURT, 2005). Porém, não é possível representar todas as formas geométricas somente por meio de vértices. Por isso, outros elementos como as arestas, proveniente da união entre dois vértices através de linha, e a união de três arestas ou mais, realizando um circuito fechado que é denominado de face (TORI CLAUDIO KIRNER, 2006), também são importantes.

### 2.3.1 Softwares de Modelagem 3D

Passamos agora a outro momento desta pesquisa, a criação de modelos tridimensionais (3D), uma tarefa nada trivial. No princípio da década de 1980 pesquisadores começaram a buscar técnicas para automatizar o processo de criação de sólidos e superfície. Um dado importante é que esses sólidos deveriam ter uma exatidão em suas medidas. Essas pesquisas da década de 1980 levaram ao desenvolvimento dos primeiros softwares de modelagem. Eles ficaram conhecidos como CAD (*computer aided design*). Sua precisão faz com que esses softwares sejam muito utilizados na engenharia (TORI CLAUDIO KIRNER, 2006). Há diversos softwares de modelagem 3D nas áreas de realismo virtual e computação gráfica disponíveis no mercado visando, sobretudo, à indústria cinematográfica. A indústria de games, simulação, dentre outras foram surgindo posteriormente. Dentre os diversos programas que estão disponíveis no mercado podemos destacar o *Autodesk 3D MAX*, *Autodesk Maya* e o *Blender* (ALVES, 2012).

O *Autodesk 3D MAX* possui como ferramentas principais a modelagem, a animação e a renderização 3D. Possui, ainda, outras diversas ferramentas, tais como, mapeamento de UV, *stereo* câmera, dinâmica e efeitos de partículas, ferramentas que auxiliam o fluxo de trabalho, ferramentas que ajudam a criação de conteúdo. Porém, o *Autodesk 3D MAX* não possui licença gratuita, sendo assim, é necessário a compra de licença para a utilização (3D MAX, 2016), uma de suas desvantagens.

Outra poderosa e completa ferramenta de animação 3D é o software *Autodesk Maya*. Ele possui ferramentas de criação de personagens, animação, dinâmicas e efeitos (partículas, cabelo, tecido, fluidos), modelagem, renderização e sombreado. Assim como o *Autodesk 3D MAX*, o *Autodesk Maya* não possui licença gratuita, sendo necessário a compra de licença para sua utilização (MAYA, 2016).

Já o pacote do programa Blender contém diversas funcionalidades, sendo a modelagem 3D a principal. Porém, outras funcionalidades também se destacam. Ele possui uma poderosa ferramenta de criação de textura de material e de animação que possibilita o processamento de iluminação estática ou dinâmica, a edição de som e, por fim, é uma excelente ferramenta de edição de vídeo. Isso quer dizer que a ferramenta Blender dá suporte a todas as etapas de produção, desde a modelagem até a pós-produção. Outro diferencial está em sua licença gratuita, não havendo a necessidade de compra de licença para a utilização (FLAVELL, 2010).

## 2.4 Motores de Jogo

Dando prosseguimento a este trabalho, pode-se notar que a complexidade da criação de jogos e simuladores impele os desenvolvedores a buscar novas ferramentas para o auxílio dessa tarefa. Atualmente, diversas ferramentas são lançadas com esta proposta.

Este trabalho elege como objeto de estudo os motores de jogo que, de acordo com Feijó (FEIJO B., 2006), são as ferramentas de mais alto nível de desenvolvimento. A eleição se justifica frente ao fato de que ele minimiza a programação, o que abstrai diversas etapas do desenvolvimento através de editores e ferramentas gráficas. Além do mais, ele possibilita grande interação entre as ferramentas de arte, softwares de modelagem e editores de imagens, tornando mais simples o desenvolvimento de uma aplicação. É importante salientar que muitos motores de jogos possuem um propósito limitado. Os exemplos mais profícuos são os motores para jogos de primeira pessoa, jogos de corrida, simulação e entre outras (CLUA; BITTENCOURT, 2005). Porém, há motores de propósito gerais. Os exemplos mais importantes talvez sejam o Unity3D e o Unreal. Eles também tentam abranger todas as áreas possíveis (BLACKMAN; WANG, 2014).

Estes são os requisitos necessários para que uma ferramenta possa ser classificada como um motor de jogo: a possibilidade de encapsular os códigos da melhor maneira possível para reutilizá-los em outros projetos; a permissão para interação entre os recursos de arte como os modelos, imagens, texturas e sons, juntamente com os recursos de programação; tornar o desenvolvimento independente da plataforma; a utilização eficientemente dos recursos disponíveis. Por fim, a permissão para o gerenciamento do projeto em todas as suas partes, também é um fator importante (FEIJO B., 2006) e (CLUA; BITTENCOURT, 2005).

Dentre os diversos modelos de motores de jogo, o software Unity3D possui características que o fazem uma ótima escolha: é um software com uma versão gratuita, o que permite exportar para diversas plataformas. Essa versão apresenta todas as funcionalidades necessárias para o desenvolvimento completo de jogos e simuladores. Sua única desvantagem é o limite de ganhos de U\$100 mil (GIBSON, 2014). Porém, se os ganhos com simuladores e jogos ultrapassar o limite de ganhos da versão gratuita, a versão paga pode ser uma boa opção. Ela apresenta algumas funções adicionais como o suporte da empresa desenvolvedora e serviços de *cloud*. O Unity3D é flexível, podendo ser utilizado em diversas plataformas adotando um único código fonte. Por fim, o Unity3D goza de grande apoio da comunidade desenvolvedora de jogos e simuladores, além de possuir uma excelente documentação (BLACKMAN; WANG, 2014).

## 2.5 Máquinas Elétricas

Seguindo o percurso desta pesquisa será trabalhada, agora, a simulação de máquinas elétricas. As máquinas elétricas são basicamente compostas de duas partes: a parte estacionária ou estator e a parte rotativa ou rotor. O estator é constituído de uma carcaça e de peças polares. Enquanto a primeira fornece um suporte físico à máquina, as segundas fornecem o fluxo magnético. As peças polares são responsáveis por distribuir

uniformemente o fluxo magnético sobre o rotor. Estes são constituídos de um eixo de aço com um núcleo que o envolve (FITZGERALD; KINGSLEY; UMANS, 2006).

As máquinas de corrente contínua possuem dois tipos de enrolamento: o enrolamento de armadura, que se caracteriza pelo fato da tensão ser induzida por meio do rotor e o enrolamento de campo, que se caracteriza pela produção de um fluxo magnético localizado no estator (CHAPMAN, 2013). Na Figura 5 encontra-se o diagrama simplificado de um motor de corrente contínua. Nessa figura pode-se observar o estator juntamente com o seu enrolamento de campo e o rotor com o seu enrolamento da armadura. Na Figura 6 é possível observar o circuito equivalente de um motor de corrente contínua em derivação, também conhecido como motor Shunt.

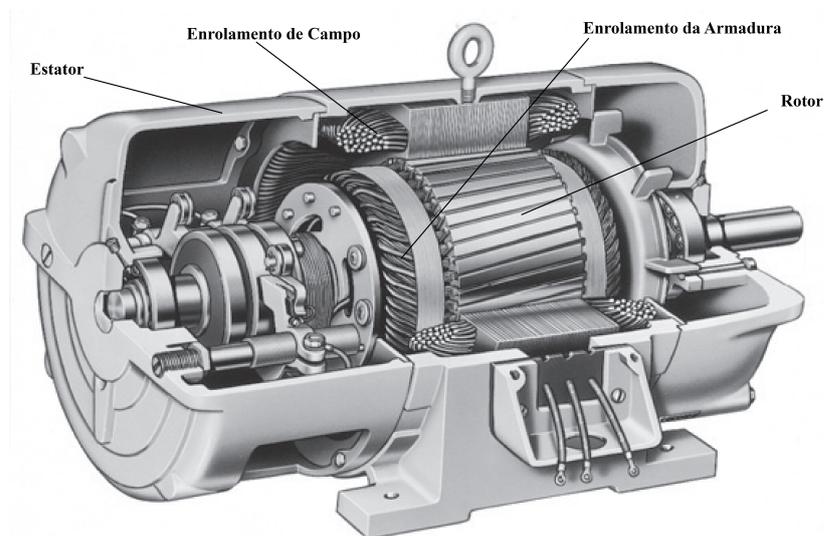


Figura 5 – Diagrama simplificado de uma máquina editado a partir de Chapman (CHAPMAN, 2013).

No motor Shunt o circuito do campo é alimentado diretamente nos terminais da armadura por uma tensão  $V_T$  (MARTIGNONI, 1971). Ao analisar esse circuito, percebe-se os seguintes componentes:  $E_A$  é a tensão ideal na armadura,  $R_A$  é a resistência da armadura,  $I_A$  é a corrente na armadura,  $I_L$  é a corrente fornecida pela fonte,  $I_F$  é a corrente no campo,  $L_F$  é o fluxo magnético gerado pelo indutor,  $R_{AJ}$  é um resistor externo ajustável,  $R_F$  é resistência do campo (CHAPMAN, 2013).

Analisando o circuito da Figura 6 pode-se obter, através da lei de Kirchhoff das tensões (LKT) seguinte equação 2.1,

$$V_T = E_A + I_A R_A \quad (2.1)$$

caso se aplique a lei de Kirchhoff das correntes (LKC) podemos obter a equação 2.2, onde  $I_L$  é a corrente fornecida pela fonte de tensão (FITZGERALD; KINGSLEY; UMANS, 2006).

$$I_L = I_A + I_F \quad (2.2)$$

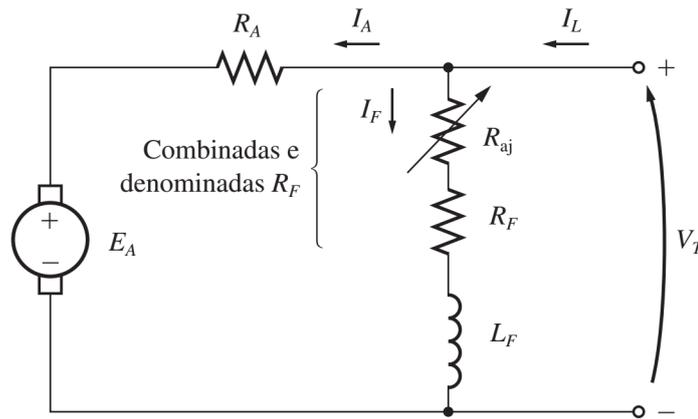


Figura 6 – Circuito equivalente de um motor de corrente contínua em derivação (CHAPMAN, 2013).

É necessário que seja definida a constante de construção da máquina, para essa função utiliza-se a equação 2.3,

$$K_F = \frac{V_{nom}}{\omega_{nom} I_{Fnom}} \quad (2.3)$$

onde  $V_{nom}$  é a tensão nominal da máquina,  $\omega_{nom}$  sendo a velocidade de rotação nominal e  $I_{Fnom}$  é a corrente nominal da mesma obtida pela equação  $I_{Fnom} = V_{nom}/R_F$ . Essas informações estão contidas na placa de identificação que se encontra fixada na máquina (MARTIGNONI, 1971).

Para determinar a tensão nos enrolamentos do rotor de uma máquina utiliza-se a equação 2.4 (CHAPMAN, 2013).

$$E_A = K_F I_F \omega_m \quad (2.4)$$

A corrente de campo, por outro lado, pode ser calculada através da lei de Ohm. É importante salientar que a observação da figura 6 é imprescindível para que seja possível obter a equação 2.5 (FITZGERALD; KINGSLEY; UMANS, 2006).

$$I_F = \frac{V_T}{R_F + R_{AJ}} \quad (2.5)$$

Através da equação 2.6 o torque da máquina pode ser definido. Nesse momento, deve-se utilizar, também, as informações nominais da máquina (MARTIGNONI, 1971).

$$T = 0,05 \frac{E_{Anom} I_{Anom}}{\omega_{nom}} \quad (2.6)$$

Por fim, é importante definir a rotação da máquina. Para tanto, utiliza-se a equação 2.7 (CHAPMAN, 2013). Com essas equações é possível realizar os cálculos necessários para a simulação.

$$\omega_m = \frac{V_T}{K_F I_F} - \frac{R_A}{K_F I_F} T \quad (2.7)$$

Pode-se controlar a velocidade da rotação do motor através de três maneiras distintas. A primeira delas é ajustando a resistência do campo  $R_F$  e como consequência alterando o fluxo no campo, a segunda maneira consiste na variação da tensão  $V_T$  aplicada nos terminais da armadura e a terceira é a menos usual, consistindo na inserção de uma resistência ajustável no circuito da armadura  $R_A$  (FITZGERALD; KINGSLEY; UMANS, 2006).

## 2.6 Considerações Finais

Neste capítulo são descritos alguns dos principais conceitos necessários para o desenvolvimento de um simulador, sendo, também, necessários para o desenvolvimento deste trabalho. A seção 2.1 aborda os processos necessários para o desenvolvimento de um simulador, com destaque para algumas metodologias, o ciclo de vida do projeto, bem como alguns profissionais envolvidos. A seção 2.2 apresenta alguns conceitos que envolvem a computação gráfica, abordados a partir da ideia de câmera virtual. Também são exemplificadas as descrições de fontes de luzes e o modo como elas interagem com a superfície do material. No final foram estudadas diferentes técnicas de aplicação de texturas. A seção 2.3 descreve a modelagem 3D e alguns softwares encontrados no mercado. A seção 2.4 aborda os motores de jogo, sendo este a principal ferramenta para o desenvolvimento deste trabalho. A seção 2.5 apresenta alguns conceitos básicos sobre o funcionamento das máquinas elétricas de corrente contínua, assim como três maneiras distintas de controlar a velocidade do motor. Com esses conceitos é possível desenvolver um simulador de atividades práticas de máquinas elétricas, bem como qualquer outro projeto relacionado a simuladores.

## 3 Pesquisas Relacionadas

Há diversos trabalhos e estudos que propõem a construção de simuladores para auxiliar o processo de ensino e a aprendizagem. Essas pesquisas buscam novas práticas pedagógicas para o ensino de diversos conteúdos. Elas se diferenciam pelos métodos, técnicas, ferramentas e tipos de projeto analisados.

Neste capítulo são apresentados alguns trabalhos cuja proposta metodológica aborda a utilização de simuladores com a função de desenvolver novos modelos pedagógicos.

### 3.1 Ambiente multimídia para o ensino de máquinas elétricas: Uma Abordagem Multidisciplinar

No trabalho Souza *et al.* (SOUZA et al., 2012) é apresentado o desenvolvimento e a concepção de um museu virtual de máquinas elétricas. Esse projeto foi desenvolvido pelo Laboratório LANTEG (Laboratório de Novas Tecnologias para o Ensino de Engenharia Elétrica) do Departamento de Engenharia Elétrica da UFRJ. Por tratar-se de um projeto multidisciplinar, foi essencial o trabalho em conjunto do departamento de Engenharia Elétrica com os departamentos de História e de Computação Gráfica. Essa união entre os cursos possibilitou uma correlação teórica entre as informações históricas e os conhecimentos técnicos.

Uma vasta pesquisa foi realizada com a intenção de buscar informações sobre os artefatos a serem digitalizados. Muitos desses dispositivos são datados entre as décadas de 1820 e 1890. Foi necessária, então, uma pesquisa minuciosa em busca de manuais da época, patentes e relatórios de exposições internacionais do período, além de papers e pesquisas feitas através da internet ou na Biblioteca de Obras Raras ou Antigas, do Centro Tecnológico da UFRJ, relatando o período. Após a pesquisa, informações e imagens foram obtidas e digitalizadas, garantindo a conservação dos documentos históricos e a democratização ao acesso dessas informações.

No museu virtual cada máquina tem sua própria página por meio da qual são detalhadas as informações acerca de seus componentes, além de uma réplica virtual que possibilita a manipulação da mesma. A réplica virtual foi construída utilizando uma tecnologia chamada de VRML (*Virtual Reality Modeling Language*). Por meio dela é possível a replicação da imagem e sua visualização online. Através deste projeto é possível oferecer uma experiência enriquecedora e estimulante aos seus visitantes. Estes têm a possibilidade de manipular as máquinas, o que seria praticamente impossível antes. Assim, o principal objetivo do museu virtual é o ensino, a aprendizagem e a divulgação científica,

auxiliando a criação de novas tecnologias para o ensino de Engenharia Elétrica.

Como resultado do trabalho que resultou no museu virtual de máquinas elétricas foram produzidos diversos artigos, *links*, imagens, máquinas elétricas virtuais e, ainda, filmes convencionais e produzidos através de realidade virtual. Todas essas informações estão disponíveis em uma página de Internet que oferece aos visitantes uma experiência incentivadora, encorajando-os a interagir diretamente com máquinas que antes só podiam ser observadas através de ilustrações.

## 3.2 Estudo de máquinas elétricas auxiliado por técnicas de realidade virtual

Tendo como principal objetivo a confecção de um sistema baseado em técnicas de realidade virtual que simule máquinas elétricas, Junior *et al.* (JUNIOR; CARDOSO; JR, 2008) apresentam uma nova abordagem para o ensino de máquinas elétricas. Ela permite melhor interpretação e visualização dos equipamentos, inclusive em partes internas.

No processo de desenvolvimento do trabalho, Junior *et al.* (JUNIOR; CARDOSO; JR, 2008) realizam, por meio de fotos e desenhos técnicos fornecidos pelos fabricantes, um levantamento dos detalhes dos componentes das máquinas elétricas. Com as devidas informações foi realizada uma modelagem tridimensional dos componentes utilizando a linguagem VRML, o que permite uma fácil replicação e visualização, inclusive através de um navegador de internet. Também foram utilizados alguns recursos como áudio, vídeo e *JavaScript* para efetuar algumas animações. Todos esses recursos foram possíveis porque o sistema permite um controle efetivo dos equipamentos desde a estrutura, até a possibilidade de desmontá-los. Ao final, foi criada uma página de internet que disponibilizou os mesmos.

Este trabalho propôs a utilização de técnicas de realidade virtual para o ensino da disciplina Máquinas Elétricas. Uma das intenções é motivar estudantes e usuários, a partir da experiência visual do funcionamento desses equipamentos, bem como realizar experimentos virtuais. Porém, na experiência de realidade virtual desenvolvida por *et al.* (JUNIOR; CARDOSO; JR, 2008) faltam informações detalhadas do funcionamento dos componentes.

## 3.3 Um Laboratório virtual tridimensional e interativo para suporte ao ensino de física

A proposta de Falcão *et al.* (FALCÃO; MACHADO, 2010) foi a criação de uma nova didática no ensino de Física. Trata-se da criação do LabSEF (Laboratório para Simulação dos Experimentos Físicos), um ambiente virtual que reproduz alguns

experimentos físicos. A intenção de Falcão *et al.* (FALCÃO; MACHADO, 2010) é auxiliar alunos e professores do ensino médio nas aulas de Física. O LabSEF possui uma versão que utiliza óculos de realidade virtual (Estereoscopia) e outra versão com uma visão tradicional, constituído basicamente por três experimentos: movimento retilíneo uniforme, queda livre e lançamento oblíquo.

Para o desenvolvimento do LabSEF foram utilizados os softwares Blender, X3D e Java. Após seu desenvolvimento ele foi disponibilizado para 33 alunos de um curso pré-vestibular. Logo após o evento teste foi aplicado um questionário de 13 perguntas. Verificou-se que 23 alunos não sentiram dificuldade, todos afirmaram que a aula se tornou mais interessante, 30 realizariam o download se disponível e 31 alunos acharam a visão 3D estimulante.

Assim, os autores concluem que o uso do LabSEF é um fator que estimula o interesse dos alunos pela aprendizagem da Física. Como proposta de trabalhos futuros, novos experimentos deverão ser inseridos, aumentando sua biblioteca. Porém, os autores não deixam clara a forma como foi disponibilizado o trabalho para avaliação dos alunos.

### 3.4 Considerações Finais

Neste capítulo são apresentadas a metodologia e a abordagem de três trabalhos que, através do uso de técnicas de computação gráfica, apresentam novas abordagens pedagógicas. Esses trabalhos além de ilustrar alguns temas abstratos como a física, também estimula o desenvolvimento do senso crítico.

Na seção 3.1 o trabalho abordado se refere ao desenvolvimento de um museu virtual de máquinas elétricas. Com o museu virtual se torna possível aos alunos interagir com máquinas elétricas tridimensionais. Anteriormente, o conhecimento sobre as máquinas elétricas só era possível por meio de ilustrações, ou claro, quando em contato físico com as mesmas. Na seção 3.2 é analisado o desenvolvimento de máquinas elétricas virtuais que possibilitam os alunos interagirem e visualizar o funcionamento das mesmas. Na seção 3.3 pode-se notar, por meio do trabalho estudado, o desenvolvimento de um simulador para o ensino de física como sendo uma nova prática pedagógica. Todos os conceitos extraídos dos trabalhos citados neste capítulo contribuíram para inspirar na elaboração e desenvolvimento da proposta desta pesquisa.

## 4 Metodologia e Desenvolvimento

Seguindo o percurso em direção à proposta deste trabalho, avaliar a possibilidade da criação de um simulador de atividades práticas para a disciplina de Máquinas Elétricas, ministrada no curso de Engenharia Elétrica, no capítulo seguinte são analisadas as metodologias aplicadas em seu desenvolvimento.

Este capítulo se divide em quatro seções. Na seção 4.1 são apresentados os recursos computacionais e ferramentas utilizadas em seu desenvolvimento. Na seção 4.2 são descritas a prática de controle de velocidade de um motor Shunt, analisando todos os passos necessários à sua realização. Na seção 4.3 são apresentados os passos seguidos no desenvolvimento deste trabalho. São abordadas as especificações do sistema, a modelagem dos objetos, a construção da cena e a atribuição dos *scripts*. Por fim, na seção 4.4 são apresentadas as considerações finais deste capítulo.

### 4.1 Recursos Computacionais e Ferramentas

Para o desenvolvimento deste trabalho vários objetos devem ser modelados tridimensionalmente. Nesse processo é utilizado o software Blender<sup>1</sup>, Ele possui muitos recursos para projetos 3D em computação gráfica e realidade virtual, oferecendo diversos recursos para a criação de objetos 3D e criação de texturas. Esses recursos também são encontrados nos softwares Autodesk 3Ds Max<sup>2</sup> e Autodesk Maya<sup>3</sup>. Porém, as duas versões dos softwares da Autodesk possuem licença de software proprietário, um diferencial no caso do Blender que não possui licença, sendo um software livre (ALVES, 2012). Pode-se observar a interface da área de trabalho do Blender na figura 7. Encontram-se aí as ferramentas de criação e edição de sólidos, bem com as propriedades do objeto.

Para obter uma maior eficiência no desenvolvimento de um jogo ou simulador utilizamos, neste trabalho, um motor de jogo que encapsula várias rotinas comuns no desenvolvimento de jogos e simuladores. O motor de jogo escolhido é o software Unity3D<sup>4</sup>, cuja principal vantagem está no suporte dado ao desenvolvimento de diversas plataformas que utilizam o mesmo ambiente. Ele também possui em sua loja uma vasta biblioteca de recursos disponíveis e uma ampla comunidade que disponibiliza de tutorias a documentos que vão auxiliar desenvolvedores iniciais e os mais experientes. Porém, seu grande diferencial é a licença gratuita que não limita o uso de suas ferramentas (GIBSON, 2014).

<sup>1</sup> Blender <<https://www.blender.org>>

<sup>2</sup> Autodesk 3Ds MAX <<http://www.autodesk.com.br/products/3ds-max/overview>>

<sup>3</sup> Autodesk Maya <<http://www.autodesk.com.br/products/maya/overview>>

<sup>4</sup> Unity3D <<http://unity3d.com/pt/>>

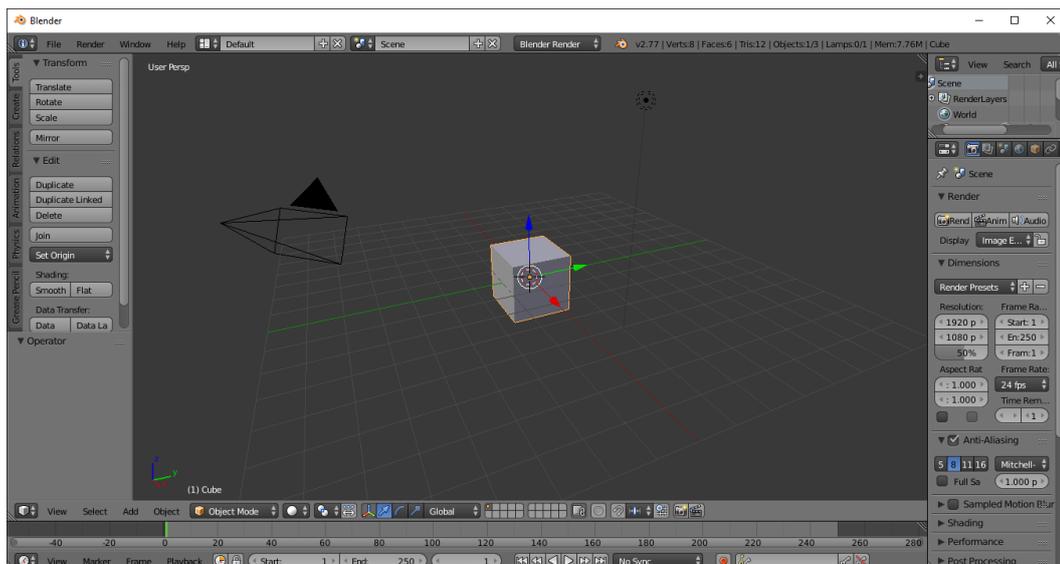


Figura 7 – Interface da área de trabalho do Blender.

Na Figura 8 é possível observar a interface da área de trabalho do Unity3D. Encontram-se aí seus diversos painéis, tais como: *Hierarchy*, que contém uma lista de todos os objetos da cena atual, *Project* no qual se encontram todos os objetos importados para o projeto, *Scene* painel no qual é possível realizar a edição dos objetos de cena, *Game* por meio do qual é possível visualizar o resultado final da cena pela perspectiva da câmera e o *Inspector* no qual encontram-se as informações e propriedades referentes ao objeto selecionado.

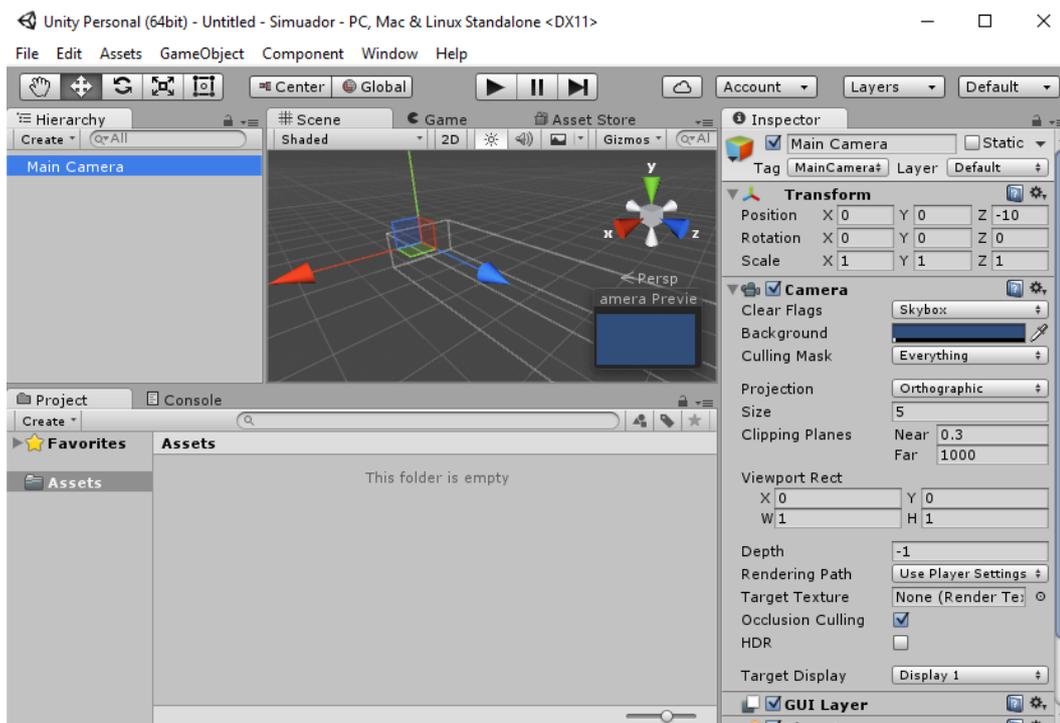


Figura 8 – Interface da área de trabalho do Unity3D.

Para tanto, é necessário o uso de um editor de texto. Neste trabalho é utilizado o MonoDevelop, que consiste basicamente em uma IDE (ambiente de desenvolvimento integrado) de código aberto disponível para Windows, Linux e OSx. Ele fornece suporte a diferentes linguagens de programação, tais como C#, Boo, Java, Nemerle, NET entre outras (MAMONE, 2006). Além disso, ele é o editor de script recomendado na instalação do Unity3D (GIBSON, 2014).

Para a utilização dessas ferramentas alguns requisitos mínimos são necessários. Eles podem ser observados na Tabela 1. Observar-se os requisitos mínimos para a utilização do Blender<sup>5</sup> e Unity3D<sup>6</sup> bem com o sistema utilizado. Observa-se que o sistema utilizado atende aos requisitos mínimos para o uso dessas ferramentas.

Tabela 1 – Requisitos mínimos para utilização das ferramentas e sistema utilizado.

Sistema Utilizado	Requisitos mínimos		
	Blender	Unity3D	
Sistema Operacional	Windows 10	Não especificado	Windows 7 SP1
Processador	Core i5 1.60GHz 64bits	Dual Core 2.0GHz 32bits	Não especificado
Memoria Ram	4GB	2GB	Não especificado
Resolução	1366x768	1280x768	Não especificado
OpenGL	OpenGL 4.5 - 2GB	OpenGL 2.1 - 512MB	Não especificado
DirectX	DirectX 12	Não especificado	DirectX 9

## 4.2 Prática do controle de velocidade de um motor Shunt

A prática escolhida para ser simulada neste trabalho consiste em observar e analisar o controle de velocidade de um motor de corrente contínua, permitindo a variação da tensão aplicada nos terminais do motor, a variação da corrente do campo ou a variação da resistência da armadura. Esta prática foi escolhida por sua quantidade de elementos visuais e, também, por ser uma das mais comuns na disciplina de Máquinas Elétricas. Para essa prática são utilizados os seguintes componentes: bancada didática; motor de corrente contínua; voltímetro de bobina móvel; miliamperímetro de bobina móvel; amperímetro de bobina móvel; cabos de conexão; carga resistiva; reostato.

Na Figura 9 é possível verificar as conexões entre o motor, a resistência do campo, a resistência da armadura, o amperímetro, o miliamperímetro, o voltímetro e a fonte de tensão. Eles devem ser configurados durante a realização da prática. Após a configuração das conexões, o trabalho consiste em aplicar na resistência da armadura ( $R_{ac}$ ) e na resistência do campo ( $R_{fc}$ ) o valor de  $60 \Omega$ . É importante lembrar que se deve aplicar o menor valor possível. Em seguida, a tensão aplicada no motor é aumentada até que o mesmo arranque e chegue a uma velocidade de 1500rpm. Em seguida, realiza-se o controle da velocidade através da variação da resistência da armadura. Mais à frente, é feito o

<sup>5</sup> Blender - Requisitos mínimos <<https://www.blender.org/download/requirements/>>

<sup>6</sup> Unity3D - Requisitos mínimos <<https://unity3d.com/pt/unity/system-requirements>>

controle da velocidade de rotação através da variação da resistência do campo. Por fim, a tensão é reduzida até que o motor pare. Todos os valores (resistência e rotação) devem ser devidamente anotados.

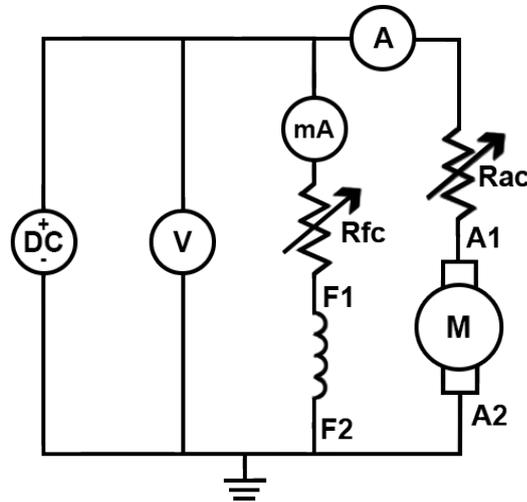


Figura 9 – Circuito de conexão do motor corrente contínua do tipo shunt.

## 4.3 Desenvolvimento

O desenvolvimento desta etapa do trabalho é dividido em quatro partes: na primeira é feito um levantamento dos requisitos, bem como a definição dos objetos que serão modelados. Na segunda etapa deve-se realizar a modelagem 3D dos objetos e a aplicação das texturas. A terceira etapa consiste na montagem da cena no Unity com os objetos modelados, aplicação da iluminação e definição da câmera. Na quarta etapa são criados os *scripts* aplicados nos objetos em cena.

### 4.3.1 Especificação do Sistema

Vários requisitos são necessários para o desenvolvimento desta pesquisa. Primeiro defini-se quais os objetos devem ser modelados. A seguir, são descritos os principais componentes que farão parte do simulador:

- Motor;
- Borne/conector;
- Cabo;
- Carga Resistiva;
- Reostato;

- Disjuntor simples;
- Disjuntor duplo;
- Disjuntor triplo;
- Botão duplo;
- Lâmpada;
- Chave de 5 posições;
- Variador da fonte;
- Variador do reostato;
- Amperímetro de bobina móvel;
- Miliamperímetro de bobina móvel;
- Multímetro de bobina móvel;
- Bancada Didática.

Na Figura 10 pode-se verificar a posição dos objetos em cena. Para tanto, é necessária a representação por meio de alguns caracteres, onde  $B$  são os bornes,  $C$  são as chaves,  $D1$  são os disjuntores simples,  $D2$  são os disjuntores duplos,  $D3$  são os disjuntores triplos,  $L$  são as lâmpadas,  $BD$  são os botões duplos,  $A$  são os amperímetros,  $MA$  são os miliamperímetros,  $V$  são os voltímetros,  $VF$  são os variadores da fonte,  $VR$  é o variador do reostato,  $M$  é o motor,  $V9$  é o reostato e  $V11$  é a carga resistiva. Na cena em questão, os cabos não estão posicionados, mas posteriormente serão.

A posição da câmera é prefixada, o que impede a livre movimentação do usuário na cena, proporcionando uma melhor visualização dos objetos em destaque. Foram determinadas 12 posições baseadas nos objetos. Na Figura 10 é possível verificar as posições das mesmas através dos caracteres que vão de  $V0$  até  $V11$ , no qual  $V0$  é a posição inicial em que a câmera se encontra afastada da bancada podendo observar todos os objetos. Em cada uma das outras marcações a câmera se movimenta até ficar mais próxima da marcação, o usuário poderá se movimentar entre elas sem a necessidade de voltar à posição inicial.

Para que todos os objetos sejam iluminados da mesma maneira definiu-se em seção prévia que a luz seja do tipo direcional, como descrito na seção 2.2.2. São atribuídos ao botão esquerdo do mouse algumas funções: ativar ou desativar um disjuntor, ativar os botões, selecionar um conector ou selecionar um cabo.

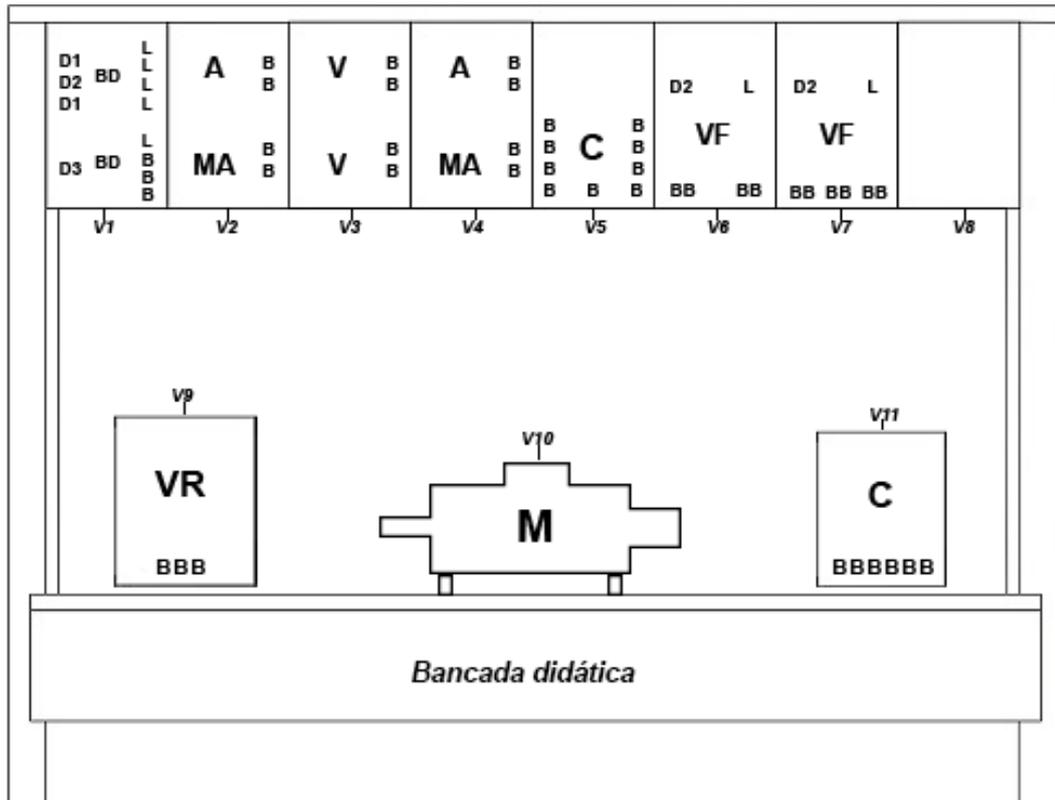


Figura 10 – Protótipo da cena com as posições dos objetos marcados por caracteres.

Também é definido que a cena é fechada, impedindo que o usuário realize ações não permitidas ou não esperadas durante a prática. Com isso, não são permitidas conexões não esperadas. Na Figura 11 são relacionados alguns números para os conectores, cuja finalidade é facilitar a relação das conexões. Na Tabela 2 pode-se verificar quais são as conexões possíveis para essa prática.

Tabela 2 – Tabela com os pares de conexões possíveis para a prática.

| Conexão |
|---------|---------|---------|---------|---------|---------|---------|
| 1       | 5       | 4       | 6       | 7       | 9       | 10 21   |
| 1       | 7       | 4       | 12      | 8       | 15      | 10 22   |
| 1       | 9       | 4       | 14      | 8       | 16      | 10 23   |
| 2       | 3       | 5       | 7       | 8       | 17      | 11 15   |
| 2       | 6       | 5       | 9       | 10      | 18      | 11 16   |
| 2       | 12      | 6       | 12      | 10      | 19      | 11 17   |
| 2       | 14      | 6       | 14      | 10      | 20      | 12 14   |
|         |         |         |         |         |         | 13 18   |
|         |         |         |         |         |         | 13 19   |
|         |         |         |         |         |         | 13 20   |
|         |         |         |         |         |         | 13 21   |
|         |         |         |         |         |         | 13 22   |
|         |         |         |         |         |         | 13 23   |
|         |         |         |         |         |         | 18 20   |
|         |         |         |         |         |         | 18 21   |
|         |         |         |         |         |         | 18 22   |
|         |         |         |         |         |         | 18 23   |
|         |         |         |         |         |         | 19 20   |
|         |         |         |         |         |         | 19 21   |
|         |         |         |         |         |         | 19 22   |
|         |         |         |         |         |         | 19 23   |

O usuário pode realizar três tipos de associações de resistores. Para essas associações são utilizados um banco de carga resistiva, elas podem ser do tipo individual, como na Figura 12a, Figura paralela 12b ou em série Figura 12c, qualquer outra forma de arranjo não é aceita.

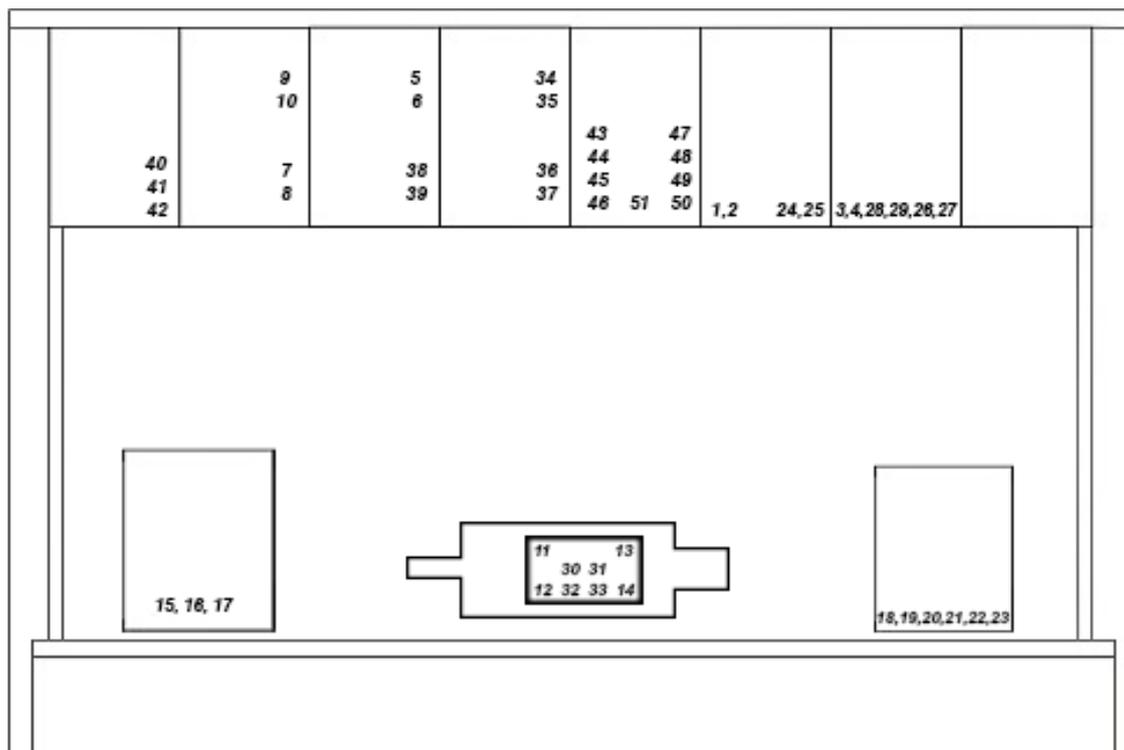
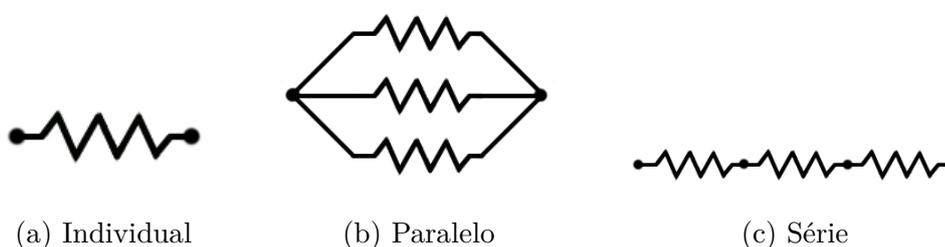


Figura 11 – Mapa com os números dos bornes.



(a) Individual

(b) Paralelo

(c) Série

Figura 12 – Tipos de associações dos resistores.

### 4.3.2 Modelagem dos Objetos

Nessa seção são abordados os passos necessários para a modelagem dos objetos 3D, para a aplicação das texturas, finalizando com a exportação para o Unity3D. Em todo esse processo utiliza-se o software de Blender.

Entende-se como *mesh* o conjunto de vértices, arestas e faces de um modelo, ou seja, de maneira geral toda área de edição e seu conteúdo. A principal técnica utilizada para a modelagem desses objetos consiste em seguir as arestas e vértices de uma imagem referência colocada em plano de fundo. Com isso, os objetos podem ser reproduzidos com perfeição. Há também um fator que ajuda no processo de modelagem: suas formas geométricas. Elas consistem basicamente em retângulos, triângulos, cilindros e esferas ou na união delas. Nessa seção é abordada a modelagem da bancada didática e a dos cabos, este último é um caso especial entre os objetos. O resultado final dos demais modelos é encontrado no apêndice A.

O primeiro passo consiste em adicionar o desenho da bancada didática (Figura 13) dentro do Blender para utilizá-la como imagem de referência. Com o Blender iniciado pressione a tecla N, observe que aparece um painel à direita (Figura 14), selecione a opção Background Imagens, logo após clique na opção *ADD Image* e selecione a imagem de referência. Com a imagem adicionada pode-se definir qual será a sua vista (*AXIS*), definindo que essa imagem será a mesma em todas as vistas. Dessa maneira, deixa-se marcado a opção *All Views*.

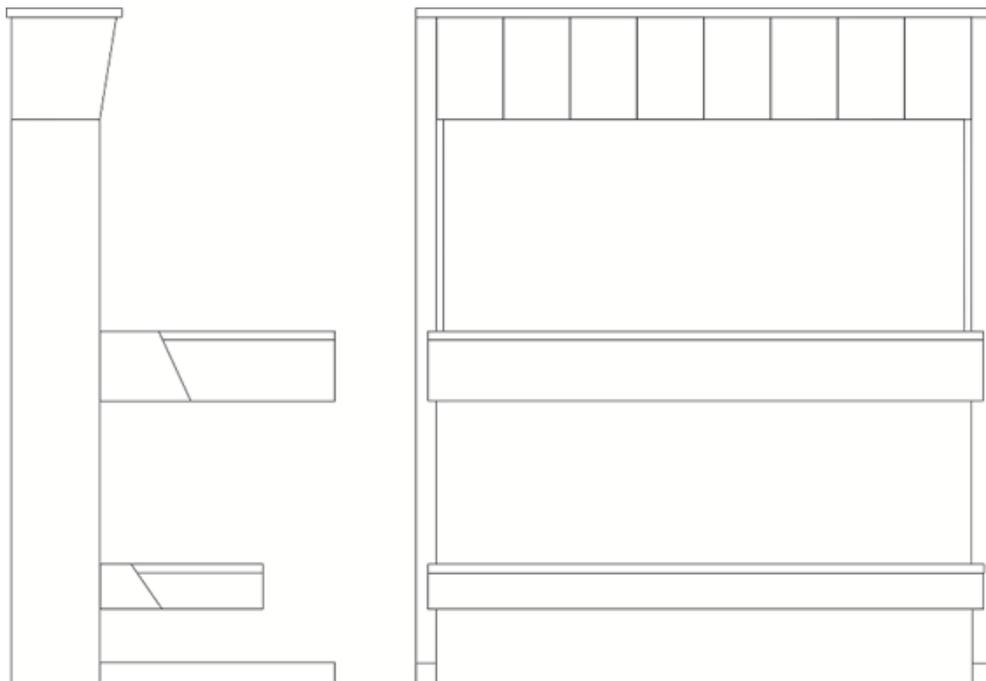


Figura 13 – Desenho da bancada didática.

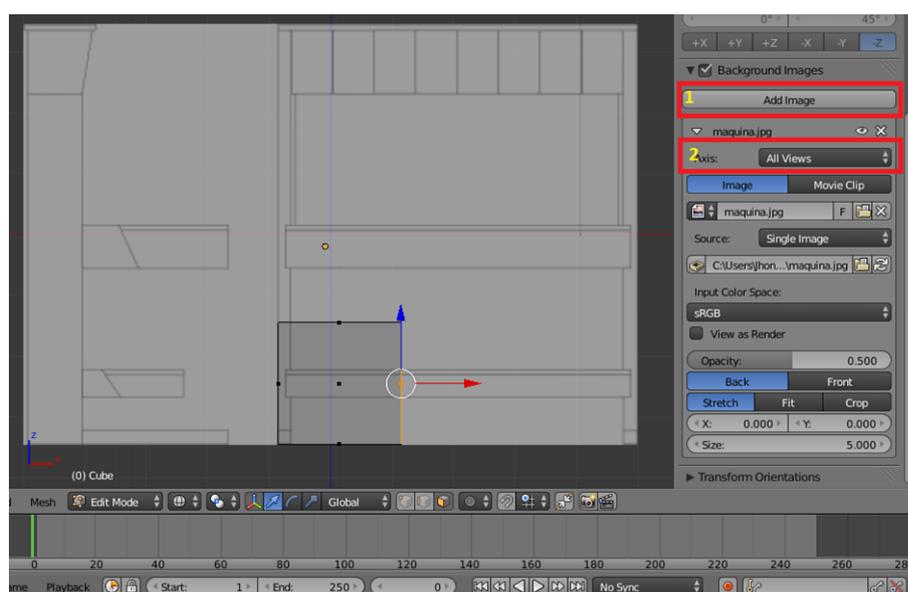


Figura 14 – Adicionando uma imagem no *background*.

Após adicionar uma imagem de fundo é possível seguir os vértices e as arestas do modelo. No entanto, primeiro é preciso trocar a câmera do tipo perspectiva (3 dimensões) para o tipo ortogonal (2 dimensões). Nesse modelo de câmera todas as linhas de projeções são paralelas ao plano de projeção, preservando a geometria do (AZEVEDO, 2003). Para realizar essa troca basta utilizar a tecla numérica 5. A primeira vista modelada é a lateral direita (*Right Ortho*), para troca da visão basta utilizar a tecla numérica 3, para realizar modificações no modelo é preciso alterar o modo de trabalho de *Object Mode* para *Edit Mode* e utilizar a tecla *tab*. Para criar um plano sobre a imagem utilize a tecla de espaço e, na janela de comando, digite *ADD PLANE*. Com o plano criado sobre a imagem ajuste o plano para cobrir toda a vista desejada, logo em seguida utilize a ferramenta de recorte (tecla K) para recortar o entorno da imagem. Algumas faces que não pertencem ao modelo aparecerão. Para removê-las bastas seleciona-las com o botão direito do mouse e, em seguida, a tecla delete. O resultado pode ser visto na Figura 15.

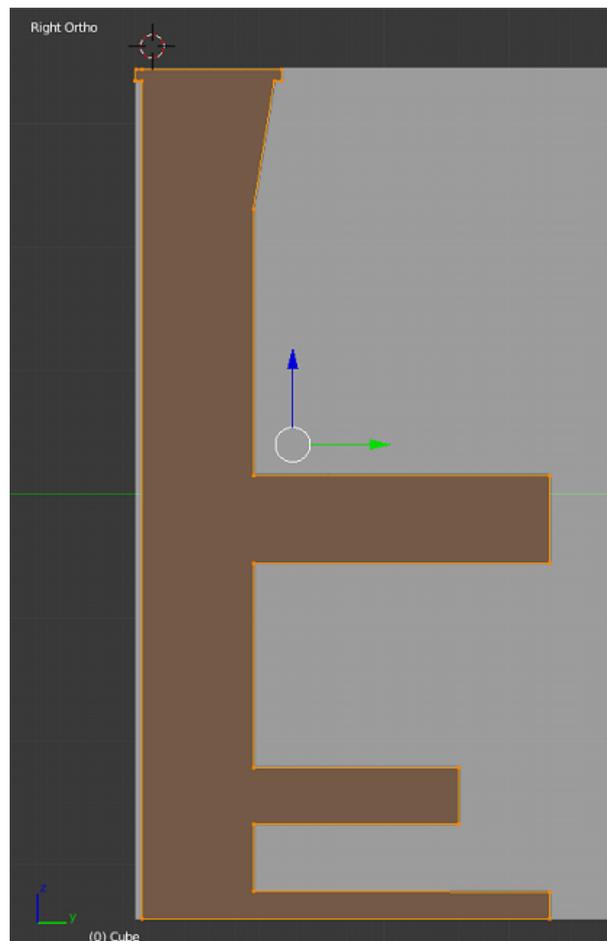


Figura 15 – Resultado da modelagem da vista lateral direita.

Com a face lateral pronta é preciso realizar a expansão do modelo para a vista frontal. Para isso, selecione a face lateral e mude a vista para a frontal (tecla numérica 1). Com o campo de visão em frontal, aplica-se a extrusão (tecla E) da face que consiste em uma expansão da face selecionada até outra extremidade. Essa parte é muito importante,

pois começa a deixar o modelo com uma aparência mais próxima à realidade. O resultado pode ser verificado na Figura 16.

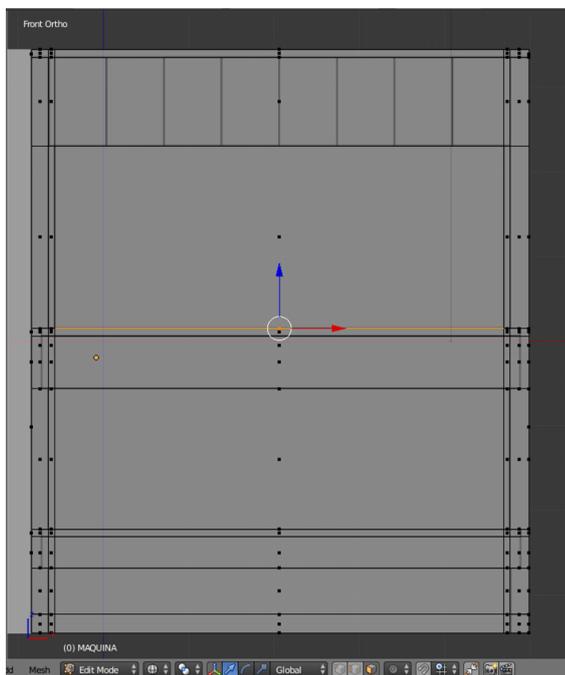
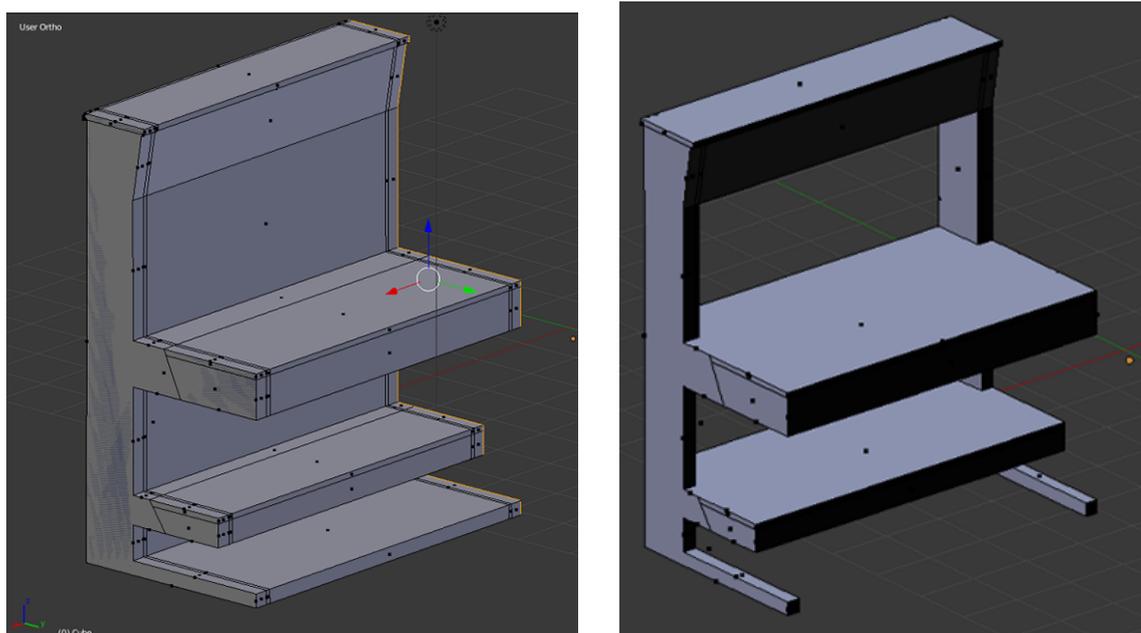


Figura 16 – Resultado da modelagem da vista frontal.

Algumas faces que não deveriam estar presentes no modelo podem ser criadas com a extrusão. Nesse modelo há um vão no centro do objeto embaixo da bancada. Porém, há faces indesejáveis e elas devem ser removidas. Assim, basta selecioná-las com o mouse e removê-las com a tecla delete. Na Figura 17a e na Figura 17b, pode-se visualizar os objetos com faces indesejáveis e sem as faces indesejáveis, respectivamente.

Após o modelo estar pronto deve-se aplicar uma textura e, especialmente nesse caso, é necessário dividir em duas texturas: uma para o corpo da bancada e outra para o painel frontal no qual estão localizados diversos elementos. Dessa maneira, torna-se necessário um maior detalhamento da textura. Mesmo após o início da aplicação da textura pelo corpo da bancada, ainda no *Edit Mode*, selecione todo o objeto, exceto a face que corresponde ao painel frontal. Pressione a tecla de espaço e, na caixa de diálogo, digite o comando *Unwrap*. Através desse comando é criado um mapa de malhas das faces do objeto conhecido como *UVs Map* (FLAVELL, 2010). Para visualizar esse mapa utilize o comando *Shift+F10*. Na Figura 18, pode-se visualizar o resultado do mapa, porém, para que seja possível atribuir-lhe cores e detalhes, é necessário exportá-lo. Para tanto, selecione o menu *UVs* e, em seguida, escolha a opção *Export UV Layout*. Como pode ser visto na Figura 19, uma nova janela é aberta permitindo que salve o *UVs map*.

Após salvar o *UVs map*, é possível visualizá-lo na Figura 20a. Com o *UVs Map* são atribuídas as cores e detalhes desejados para o modelo. Nesse processo é preciso utilizar um editor de imagens tais como o *MS paint*, *Gimp* e tantos outros softwares disponíveis no mercado. Na Figura 20b pode-se observar o *UVs Map* após a edição de cores e detalhes,



(a) Objeto com as faces indesejáveis

(b) Objeto sem as faces indesejáveis

Figura 17 – Bancada didática logo após a extrusão da face lateral.

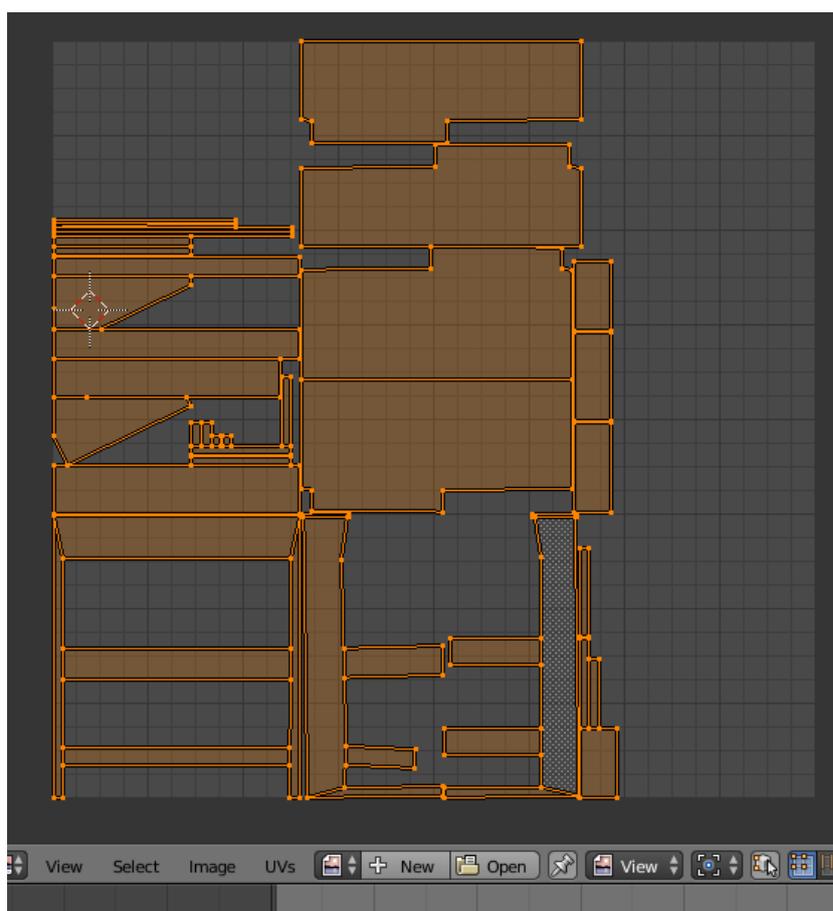


Figura 18 – Resultado da Unwrap do objeto da bancada didática.

porém essa textura deve ser aplicada ao objeto. Para isso, basta voltar à janela de edição de *UVs Map* (Figura 18) e escolher a opção *Open*. Após inserir a imagem deve-se retornar à janela *3D View* (Shift+F5) e, assim, visualizar a textura aplicada. Porém, ainda falta a textura do painel. Para isso, basta selecioná-lo e repetir o mesmo processo realizado para a bancada. Ao final, é necessário aplicar duas texturas em um objeto e, em seguida, pode-se visualizar o resultado da aplicação das texturas na Figura 21.

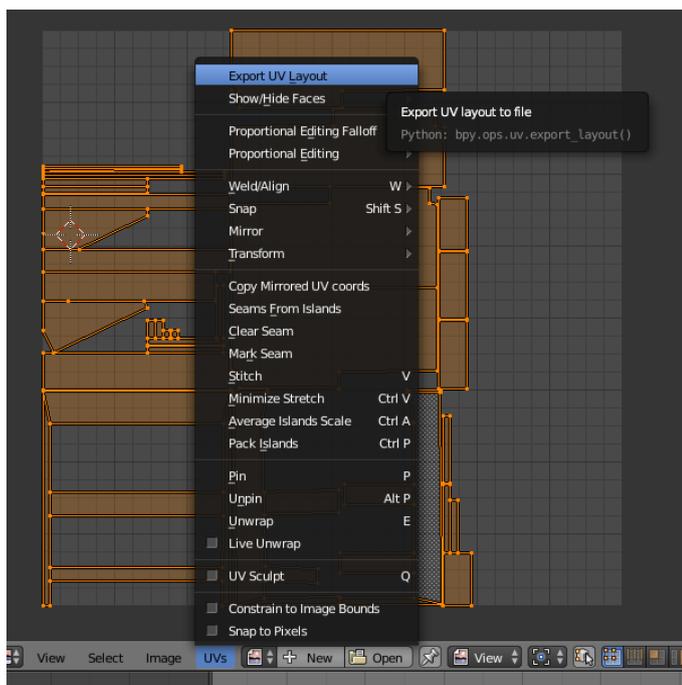


Figura 19 – Exportando o mapa de textura.

Após a aplicação da textura é necessário exportá-lo para que ele possa ser importado no Unity3D. Para isso, selecione o *menu file* que fica no canto superior à esquerda e escolha a opção exportar e, em seguida, *fbx*. Agora, basta escolher o destino do arquivo. Recomenda-se que a textura e o arquivo exportado estejam na mesma pasta.

Praticamente todos os objetos podem ser modelados com a técnica de imagem de referência. Porém, há uma técnica que une o uso de cilindros com as curvas de Bézier (curvas definidas através de um ou mais ponto de controle) (AZEVEDO, 2003) e modela os cabos de forma mais eficiente e melhor. Para o trabalho com essa técnica é necessário que os objetos estejam em cena, já que só assim é possível utilizar as posições das conexões para criar os cabos. Dessa maneira, é necessário que os objetos sejam importados para cena. Para isso, utilize o *menu file, import* e escolha a opção *fbx*, e importe os seguintes objetos: bancada didática, reostato, carga resistiva e o motor posicionando-os em cena.

Essa técnica pressupõe que a curva de Bézier dará forma ao cilindro, já que alterando a forma da curva de Bézier, altera-se a forma do cilindro. Começa-se adicionando um cilindro na cena, para isso basta digitar o comando *ADD Cylinder* na caixa de comando (tecla espaço). Em seguida, no painel de transformação (tecla N), são realizados alguns

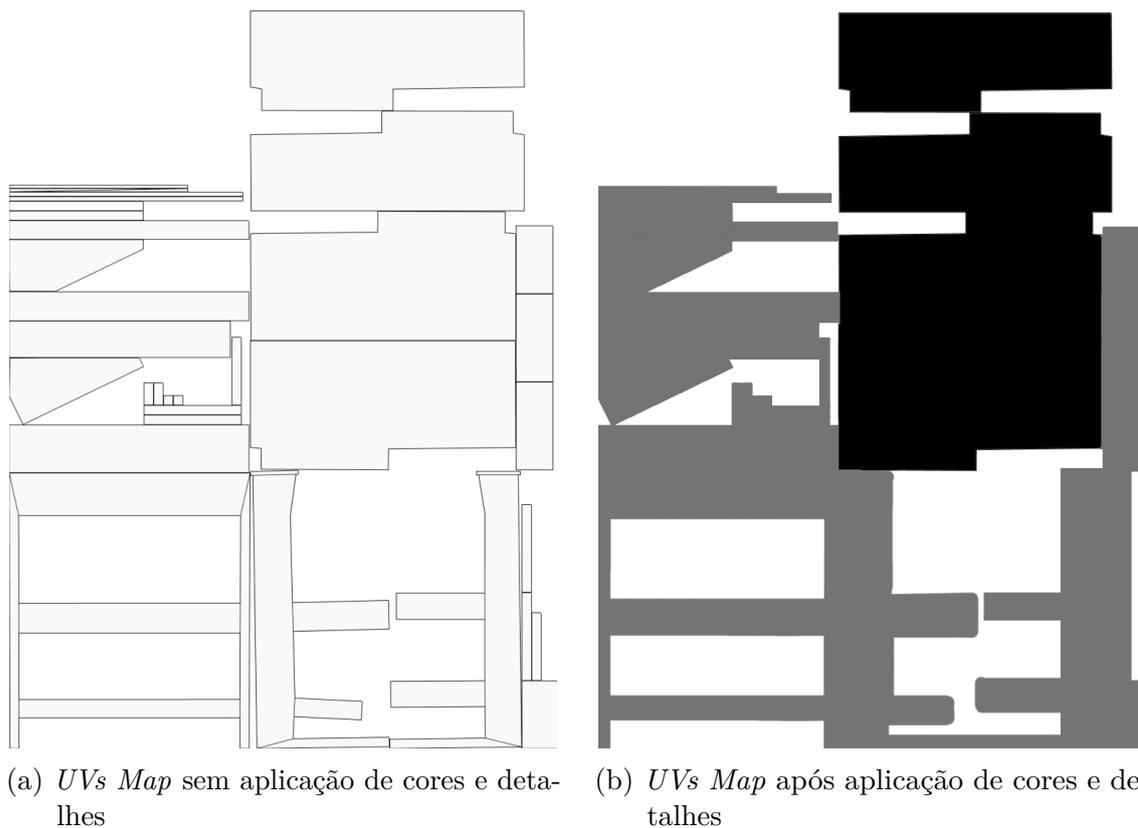


Figura 20 – *UVs Map* da bancada didática.

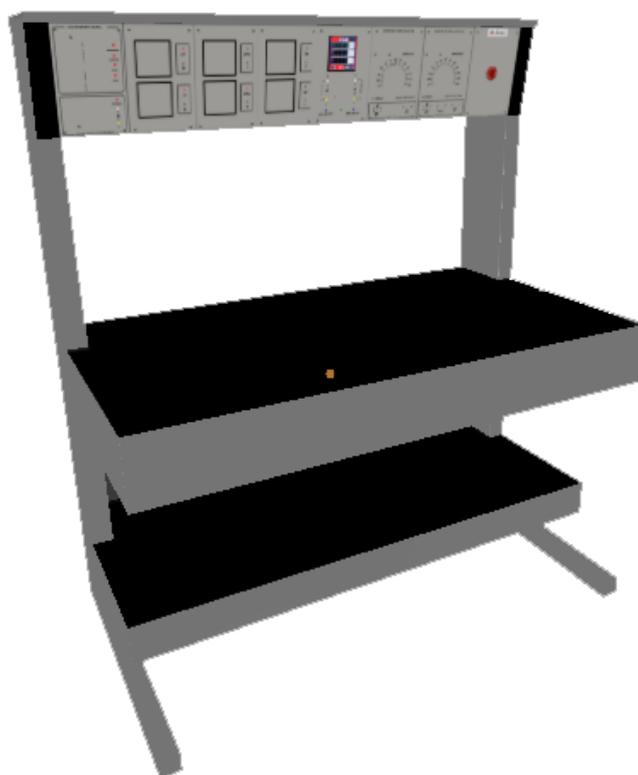


Figura 21 – Modelo da bancada didática após aplicação de textura.

ajustes na sua escala. O valor utilizado para esta pesquisa foi de 0,05 na *scale* e 0,1 na *dimensions*, com isso se consegue uma seção do cilindro. É interessante que essa seção seja pequena. Para realizar uma curva é necessário dezenas dessas seções. Para criar outras seções realize a extrusão (tecla E) da face frontal do cilindro. Sua quantidade e tamanho determinam a suavidade da deformação do cilindro pela curva. Nesse projeto foram utilizadas 50 seções.

Após a criação do cilindro, uma curva de Bézier deve ser desenvolvida. Para tanto, basta utilizar o comando *ADD Bezier*. Na caixa de comando (tecla de espaço) a curva é representada por uma linha, altere o modo de trabalho para *Edit Mode*. Desse modo, é possível observar os pontos de controle. Como podem ser observados na Figura 22, eles estão representados pela cor amarela. Com esses pontos é possível realizar a inclinação das curvas. Os pontos de controle de comprimento são representados pela cor vermelha, com eles é possível determinar a posição inicial e final da curva, e realizar a expansão ou retração da mesma. Esses pontos são sempre compostos por um par de pontos de controle de curvas. Para adicionar novos pontos de controle de comprimento, basta selecioná-los e realizar a extrusão (tecla E) e um novo ponto surgirá juntamente com dois pontos de controle de curvas, criando novos pontos de comprimento. Seguindo esse percurso é possível realizar as curvas de maneira mais suave.

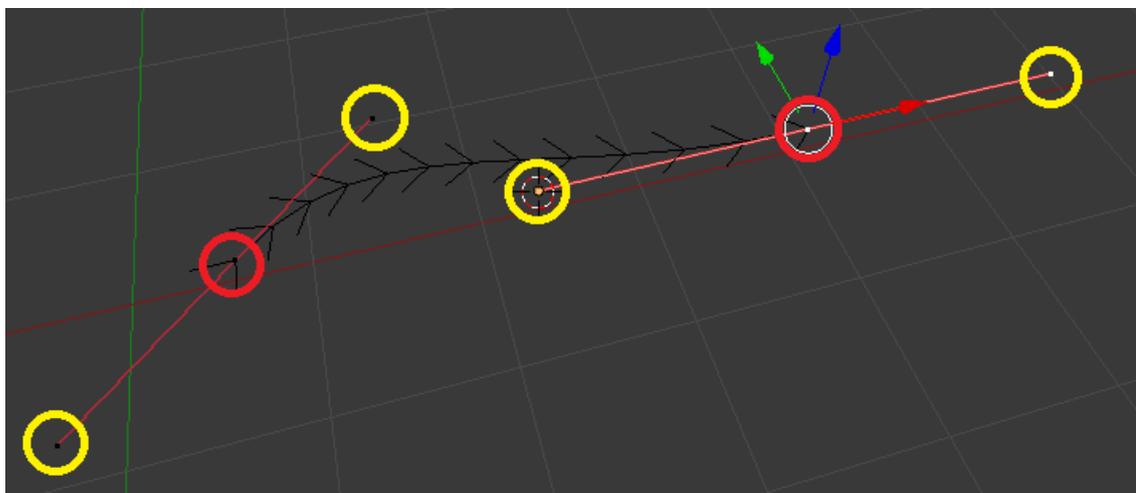


Figura 22 – Curva de Bézier no Blender, os pontos de controle da curvas são representados pelo cor amarela, os pontos de comprimento são representados pelo cor vermelha.

Com o cilindro e a curva de Bézier criados é necessário conectá-los. Para isso, basta adicionar um *modifier* ao cilindro. Com o cilindro selecionado, escolha a opção *modifier* no editor de propriedades e, em seguida, *ADD Modifier* (Figura 24a). No menu *modifiers* (Figura 23) há diversas opções de *modifiers*, selecione a opção *curve* e, retornando ao editor de propriedades, escolha a opção *Object* e selecione o objeto referente à curva. No final desse processo há uma curva de Bézier que modifica a forma do cilindro.

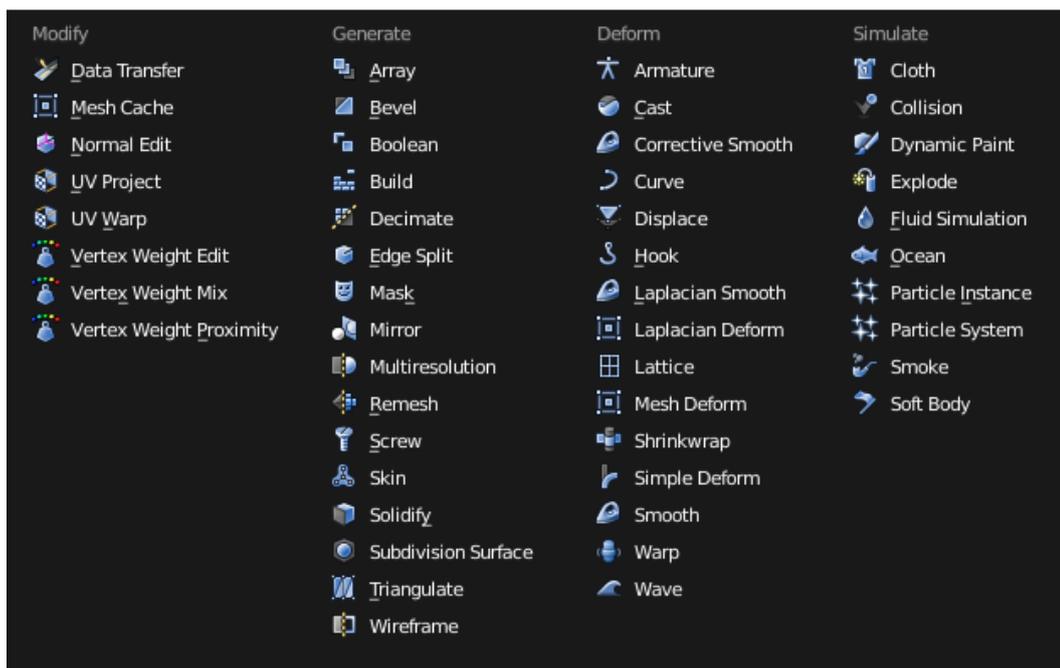
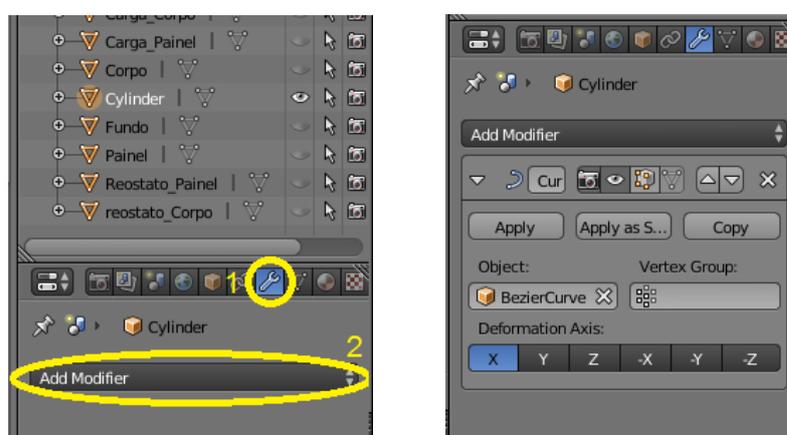


Figura 23 – Menu com os *modifiers*.

Com base na tabela 2, que consiste nos pares de conexão, e a Figura 11, que são referências das posições dessas conexões, posicione os pontos de controle de comprimento nas posições das conexões e ajuste a curvas de acordo com a necessidade. Um cilindro e uma curva de Bézier devem ser criados para cada uma das conexões e assim, ao final, têm-se todos os cabos necessários para o projeto. Como não são mais necessários, pode-se remover a bancada, reostato, carga e o motor e, por fim, o objeto é exportado. Na Figura 25 pode-se observar o resultado da modelagem de todos os cabos juntamente com os objetos de referência.



(a) Adicionando um *modifiers* ao cilindro

(b) Atribuindo um *Bézier* ao objeto

Figura 24 – Atribuindo uma curva *Bézier* ao cilindro.

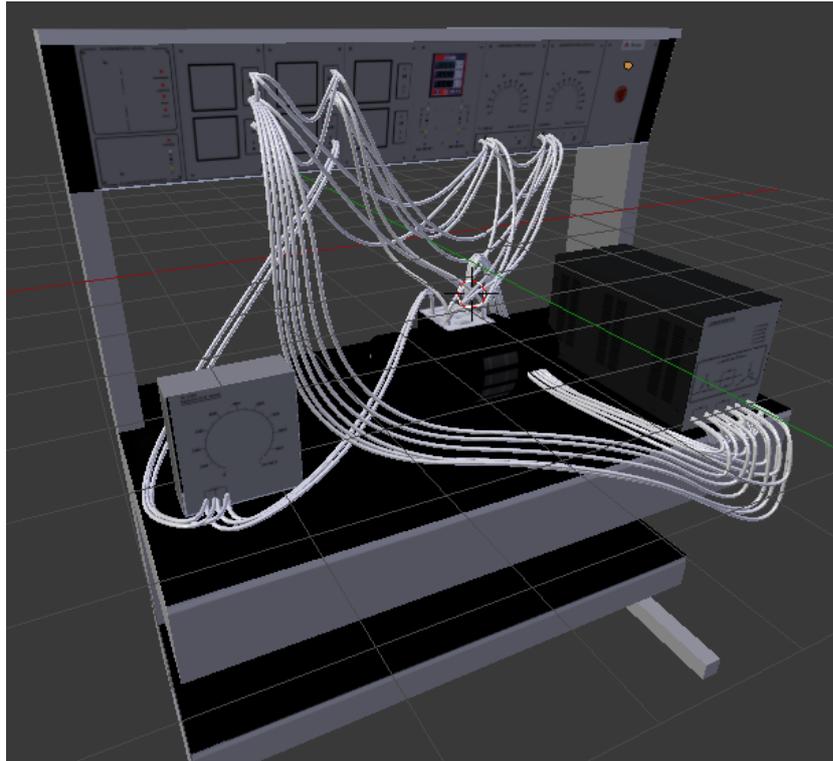


Figura 25 – Resultado da modelagem dos cabos juntos com os demais objetos.

### 4.3.3 Construção da Cena

A próxima etapa consiste na montagem da cena utilizando os objetos modelados. Esse processo está dividido em duas cenas: menu inicial e prática. Porém, é necessário se familiarizar com a interface do Unity. Pode-se observar através da Figura 26 os painéis que constituem sua interface de área de trabalho.

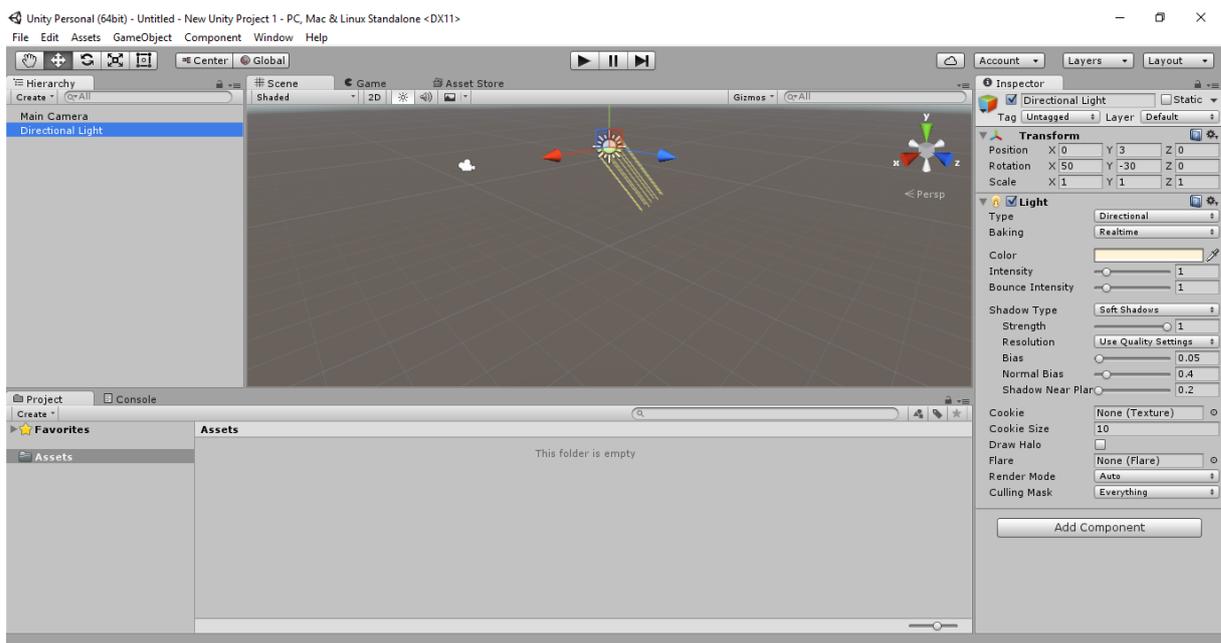


Figura 26 – Painéis da área de trabalho Unity3D.

Observa-se no canto à esquerda o painel de *hierarchy*, nele estão contidos os objetos que compõe a cena atual. À direita pode-se observar o painel de *inspector*, nele se encontram as propriedades referentes ao objeto selecionado. Dessa maneira, as informações que compõem o painel dependem diretamente do objeto selecionado. Na parte de baixo se encontra o painel de *project* juntamente com o painel de console. No painel de *project* estão contidos todos os arquivos utilizados na confecção do projeto independentemente se são utilizados ou não em alguma cena. No painel de console se encontram os relatórios de compilação, possíveis erros e a saída de texto para o *debug*. Por fim, no centro se encontram o painel de *scene* e o painel de game. No painel *scene* os objetos são visualmente localizados. Nele é possível selecionar um objeto e editar sua posição, orientação e escala, também é possível alterar a visualização da cena livremente. No painel de game se encontra o resultado da cena de acordo com a posição da câmera, objetos e iluminação. Para visualizar o resultado final da cena basta clicar no botão *play* que se encontra acima do painel juntamente com os botões *pause* e *stop*.

Todos os componentes ou objetos dentro da cena no Unity são chamados de *gameObject*. Há dois *gameObject* básicos que precisam ser criados: a câmera e a iluminação. Para inserir a câmera basta ir ao menu *gameObject* e selecionar a opção câmera. Para inserir uma iluminação basta ir novamente ao menu *gameObject*, em seguida *light*, e selecionar o tipo de iluminação. Para que a cena possa ser construída os objetos modelados devem ser importados para dentro do Unity. O Unity possui várias formas de uso que possibilitam a importação de conteúdos para a sua área de trabalho, em uma delas os arquivos podem ser colocados diretamente na pasta do projeto. Nesse processo, utiliza-se o sistema de arquivo do seu sistema operacional *...nome do projeto/Assets/* que automaticamente será atualizado dentro do Unity. Por fim, pode-se utilizar o sistema de clique e arraste selecionando um arquivo do computador e arrastando para dentro das pastas do painel *Project*.

Com os objetos devidamente importados, precisa-se montar a cena. Para isso, utiliza-se o sistema de clique e arraste selecionando um modelo dentro do painel *project* e arrastando-o para dentro do painel *hierarchy* ou *scene*.

Há um componente necessário para que haja interação entre os objetos e o usuário: esse componente é o *collider*. Ele é responsável pela verificação da colisão entre os objetos. Há várias opções de *collider* (Figura 27) dentre elas estão o *Box Collider* (Figura 27a), *Capsule Collider* (Figura 27b), *Sphere Collider* (Figura 27c) e o *Mesh Collider*. Cada um deles desenvolve um volume em volta do objeto. São respectivamente: uma caixa, uma esfera, uma capsula, e o último desenvolve um volume semelhante ao do objeto. Para inserir esse componente no objeto, selecione-o e vá ao *menu component* e escolha *physics*. Dentre as opções de *collider* selecione a *Box Collider*. Ainda com o objeto selecionado, observe o componente *Box Collider* que se encontra no painel *inspector*, nele você poderá

alterar o tamanho e posição do volume de colisão. Esse processo deve ser repetido em todos os objetos que recebem alguma interação.

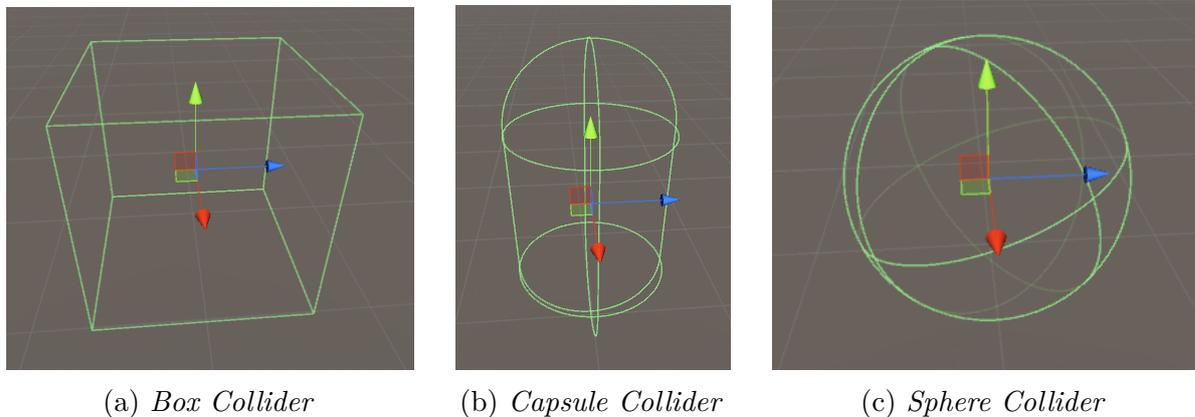


Figura 27 – Algumas opções de *collider* no Unity.

Outros componentes importantes são os de interface, necessários para orientar os usuários. Para isso, há os *User Interface (UI)*. Com esse sistema é possível a criação rápida de interfaces de usuário. A *UI* respeita a hierarquia para renderização dos componentes na tela. A hierarquia superior é a *UI CANVAS*, nela devem ser agrupados todos os demais *UI*.

Alguns desses componentes são os *UI Button*, *UI Text*, *UI Image*, dentre outros. Porém, dos utilizados neste projeto, o *UI Button* tem a função semelhante a um botão, possibilitando ao usuário clicá-lo e, também, que o sistema trabalhe essa interação. O *UI Text* nos permite a inserção de uma caixa de texto, e com *UI Image* é possível inserir uma imagem na interface.

#### 4.3.4 *Scripts*

Com os objetos em cena, deve-se construir a interação entre eles. Para isso, deve-se inserir um componente *script* aos objetos. O *script* é um código fonte atribuído a um *gameObject*. Ele é indispensável para que seja possível responder às interações do usuário e realizar os eventos previstos nos projetos. Não há um *script* principal com atributo de *main*, ele está relacionado diretamente a algum objeto em cena.

Na Figura 28 é possível verificar o painel de *Inspector* do componente variador do reostato no qual se verifica a atribuição de um *script*. Este é responsável pelo controle de rotação do variador, bem como, o cálculo do valor do reostato. Através dos *scripts* e das equações apresentadas na seção 2.5, são realizados os cálculos da tensão, corrente e velocidade apresentados aqui.

As interações e os resultados podem ser provenientes e observados através de vários objetos, sendo possível o mesmo *script* ser atribuído a diferentes objetos e apresentar



Figura 28 – Painel do *Inspector* com as informações do variador de restado.

diferentes interações e resultados. Há duas linguagens que podem ser utilizadas em um projeto no *Unity*: o *C#* e o *JavaScript*. É importante ressaltar que ambas podem ser utilizadas no mesmo projeto. Além disso, é possível atribuir um *script C#* e *JavaScript* simultaneamente ao mesmo objeto.

## 4.4 Considerações Finais

Apresentam-se, neste capítulo, as metodologias aplicadas para o desenvolvimento do simulador de atividades de práticas para a disciplina de Máquinas Elétricas. São abordados os passos adotados para o planejamento e desenvolvimento desta pesquisa, bem como, são expostas todas as suas etapas: da concepção até o resultado final.

A seção 4.1 apresenta os recursos computacionais e ferramentas utilizadas. A seção 4.2 descreve passo a passo a prática do controle de velocidade de um motor Shunt. A seção 4.3 apresenta os passos para desenvolvimento desta pesquisa, abordando as especificações do sistema, modelagem dos objetos, construção da cena e atribuição dos *scripts*.

## 5 Resultados

Neste capítulo são apresentados os resultados obtidos após o desenvolvimento desta pesquisa. A seção 5.1 descreve as interfaces elaboradas. A seção 5.2 descreve seu funcionamento, apresentando os passos necessários para a realização da prática e obtenção dos resultados, comparando-os com os valores reais obtidos em laboratório. Por fim, a seção 5.3 apresenta as considerações finais do capítulo.

### 5.1 Interface

Nesta seção são apresentadas as interfaces desenvolvidas para esta pesquisa. Elas têm por finalidade providenciar facilidades ao usuário e ser intuitiva para ele. Uma interface de menu inicial foi desenvolvida para que o usuário seja introduzido ao simulador e tenha a possibilidade de escolher a prática a ser simulada. Isso também possibilita futuras inserções de novas práticas. Na Figura 29 é possível visualizar a interface do menu inicial.



Figura 29 – Interface do menu inicial.

Ao iniciar a cena correspondente à prática, algumas interfaces são inseridas para melhor esclarecimento do usuário. Ao inserir uma *UI* no canto superior à direita, a *UI* corresponde uma *UI Image* que, por sua vez, representa o circuito utilizado. Quando o usuário realiza uma conexão bem sucedida, a imagem apresenta um realce no trecho representado pela conexão. Essa pode ser observada na Figura 30. Abaixo do circuito

encontram-se uma UI em forma de caixas de textos que informam o status de diversos componentes. Sua finalidade é aferir resultados tais como: a tensão no voltímetro, as correntes nos amperímetros, os valores de resistências aplicadas no reostato e na carga resistiva e a velocidade de rotação do motor de maneira mais precisa. Tais informações se encontram na Figura 30.

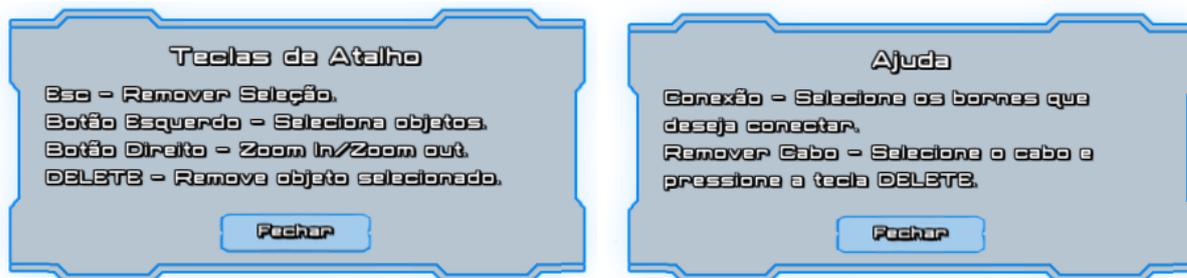


Figura 30 – Interface da cena principal.

Na parte inferior, encontra-se uma UI representada por uma caixa de texto que reporta qual o status da simulação. Este pode ser: operações bem sucedidas, erros cometidos e instruções de como devem ser a utilizados, mensagens do tipo Conexão não permitida, Clique com o botão esquerdo sobre o cabo para seleciona-lo, Todos os equipamentos estão conectados, dentre outras.

Por fim, no canto superior à esquerda, encontra-se a UI do menu. Nela, encontram-se várias configurações e informações para o usuário: uma janela com as teclas de atalho 31a) uma janela com explicação de como realizar as conexões e remover os cabos (Figura 31b). Há, ainda, opções que auxiliam o usuário na exibição ou ocultamento das informações de status e do circuito. Além de tudo isso, é possível retornar ao menu inicial ou encerrar a simulação.

Durante o desenvolvimento, devido à precisão do movimento circular do mouse, foi detectado uma dificuldade na implementação do controle de variação de tensão e do reostato. Para solucionar esse problema foram inseridos dois botões (mais e menos) ao lado dos variadores, possibilitando ao usuário aumentar ou diminuir o valor dos mesmos. Na Figura 32 podem ser observados ambos os casos. É importante destacar que ao clicar

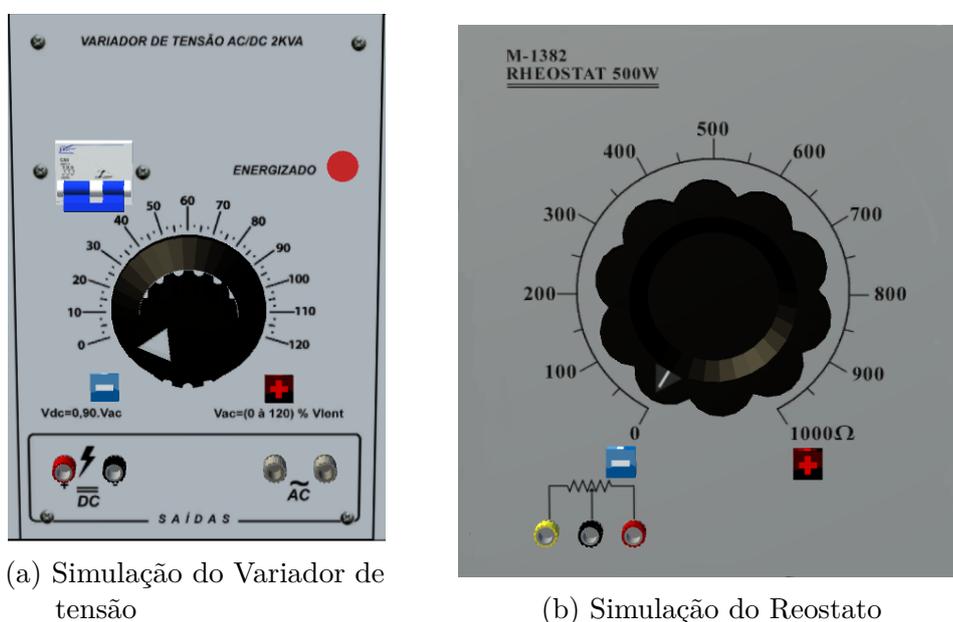


(a) Janela com as teclas de atalhos

(b) Janela com opções de ajudas

Figura 31 – Janelas de ajuda ao usuário.

nos botões sua cor será alterada para vermelho, simbolizando sua ativação. Ao soltar o botão sua cor original é retornada.



(a) Simulação do Variador de tensão

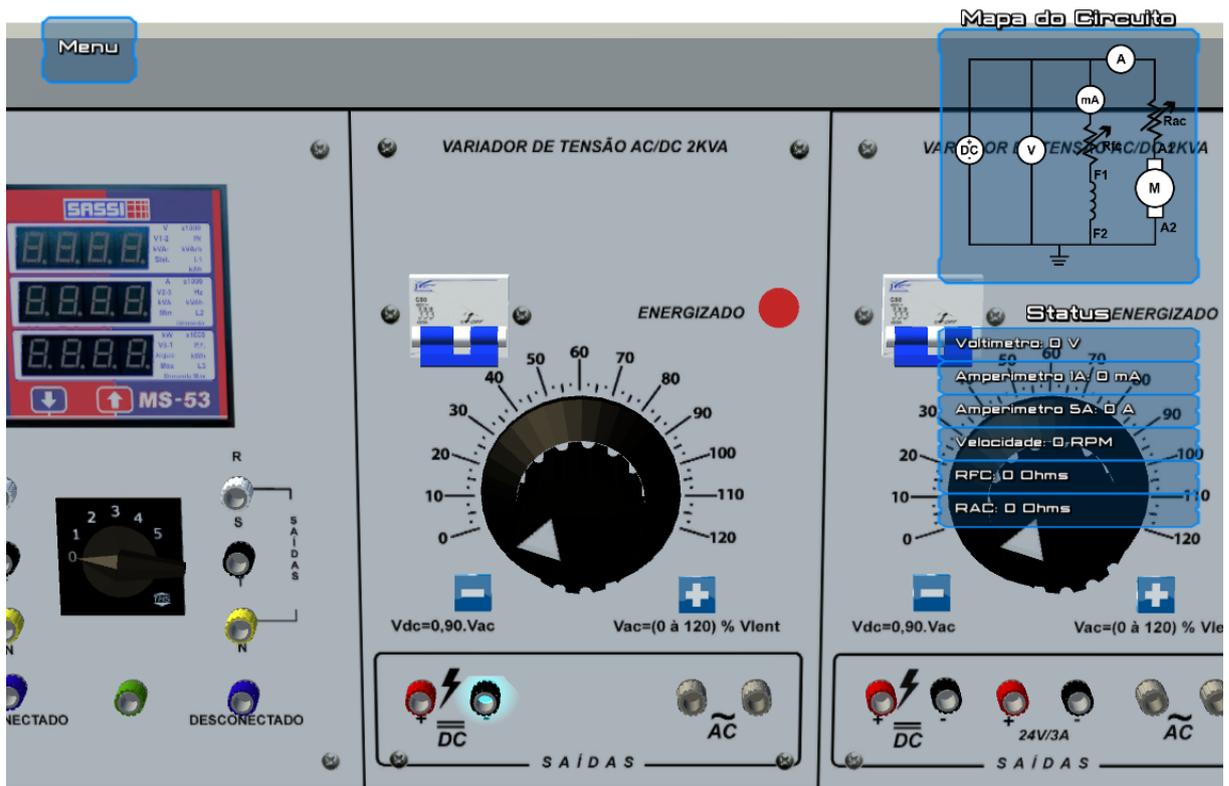
(b) Simulação do Reostato

Figura 32 – Objetos com variador

Com as interfaces devidamente colocadas em cena, quando o usuário realiza a seleção de cabo ou conector, são inseridos detalhes de realce em objetos. Eles ajudam o usuário a compreender que a ação de selecionar os objetos foi bem sucedida. Nas figuras 33 e 34 esses realces podem ser observados.

## 5.2 Funcionamento

Após o desenvolvimento, foram realizados testes com a finalidade de verificar suas funcionalidades e os resultados da simulação. Devido às conexões pré-definidas que a simulação possui, nenhuma outra conexão poderá ser realizada. Porém, mesmo que a conexão não possa ser realizada, todos os conectores devem ter a possibilidade de serem



Para conectar as fonte em série, conecta o NEGATIVO da fonte 1 com o POSITIVO da fonte 2

Figura 33 – Realce de um conector devido a sua seleção.

selecionados. Durante os testes, nenhuma conexão não esperada foi realizada e todos os conectores puderam ser selecionados.

Cada motor possui informações referentes ao seu funcionamento, tais como tensão nominal, rotação nominal e corrente de campo nominal. Para essa simulação utilizam-se respectivamente os seguintes valores: 180V, 1800rpm e 1A. Há várias formas e ordens possíveis para realizar as conexões entre os componentes. Começam-se ligando as duas fontes DC em série, para que seja possível obter a tensão nominal de 180V. Em seguida, liga-se a fonte ao voltímetro. Porém, ele não é necessário para o funcionamento do motor. Este será ligado para que seja possível visualizar a tensão aplicada. Essas conexões podem ser verificadas na figura 35. Um realce no mapa do circuito, à direita da cena, pode ser observado. Ele representa a fonte e o voltímetro conectados corretamente. A mensagem *Fonte Conectada – Série* pode ser lida, reportando ao usuário o tipo de conexão da fonte.

Após a conexão da fonte e do voltímetro, liga-se o miliamperímetro e o reostato no enrolamento de campo do motor. Essas novas conexões se referem ao fluxo de campo. Conecte o miliamperímetro à fonte e ao reostato. Há uma peculiaridade na conexão do reostato, já que os dois conectores da extremidade, quando conectados fornecem o valor total de 1000Ω. Porém, ao utilizar um conector da extremidade e o central pode-se controlar

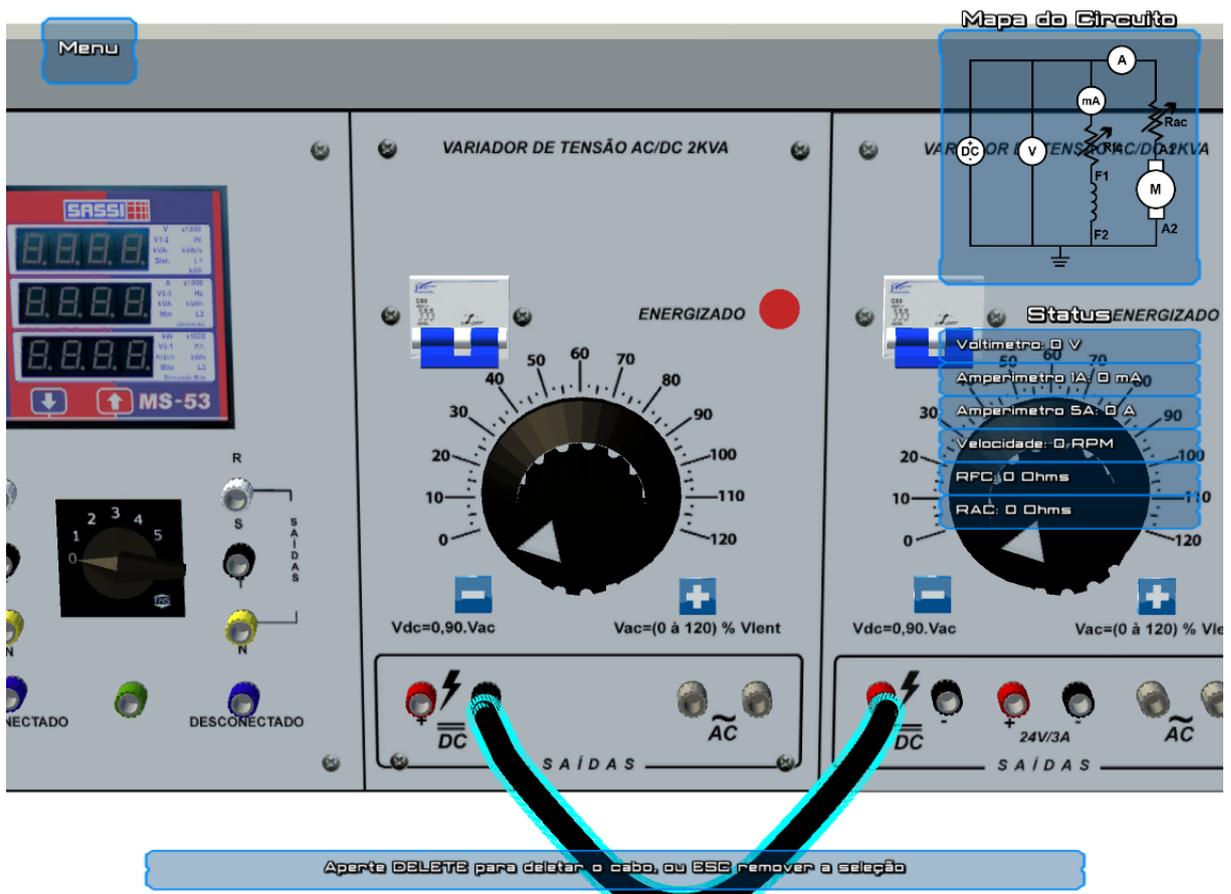


Figura 34 – Realce de um cabo devido a sua seleção.

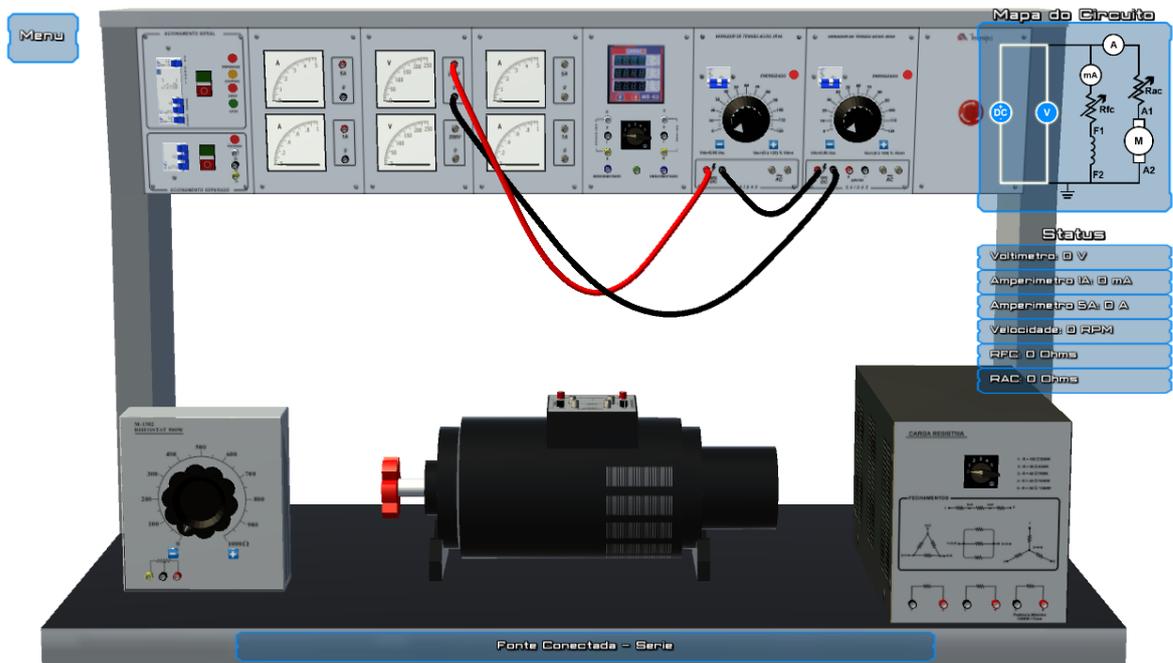


Figura 35 – Simulação com a fonte em série e o voltímetro.

o valor fornecido pelo reostato. Como essa prática requer um ajuste no valor fornecido, utiliza-se essa configuração para, então, em seguida conectar o reostato ao motor através

do conector F1 do motor. O conector F2 do motor deve ser conectado ao conector de referência (negativo) da fonte. Observar-se na figura 36 essas conexões, juntamente com a fonte e o voltímetro. Tudo isso pode ser observado no mapa do circuito no qual o realce representa os componentes conectados.

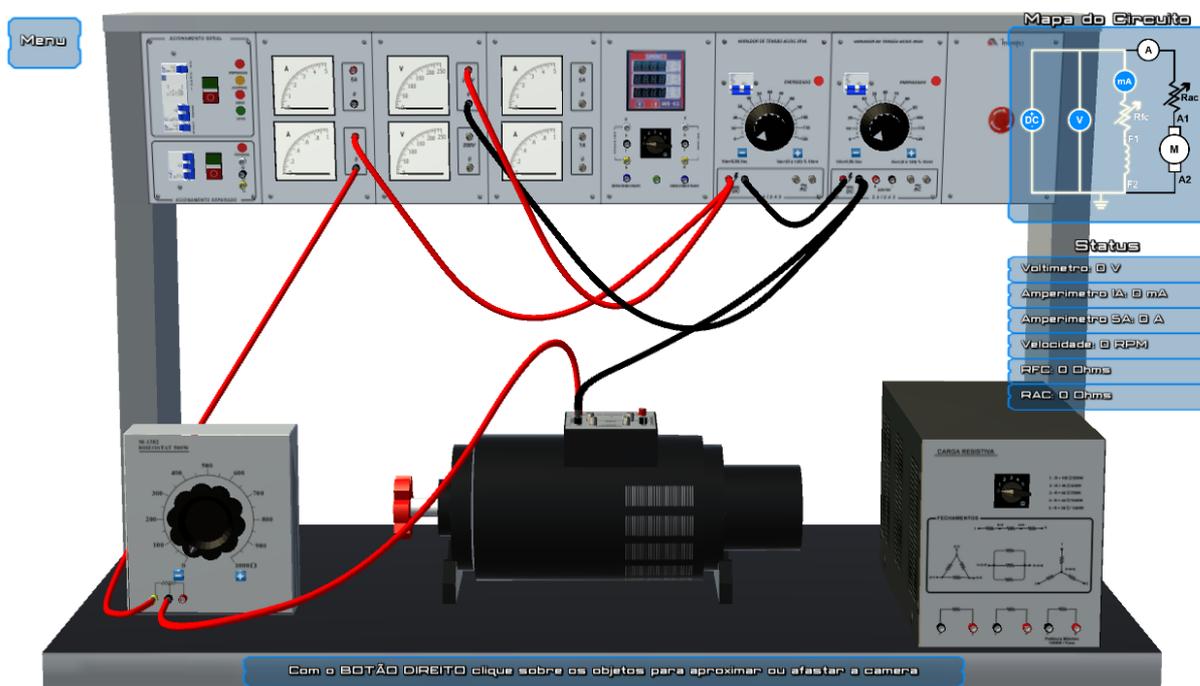


Figura 36 – Simulação com a fonte em série, voltímetro, miliamperímetro, reostato e o enrolamento de campo.

As últimas conexões são referentes à corrente da armadura. Conecte o amperímetro à carga resistiva e ao enrolamento da armadura do motor. Há três maneiras distintas de configurações da carga resistiva: individual, série e paralelo. Para esse exemplo, é utilizada a configuração em paralelo. Para isso, conecte entre si os conectores pretos e, em seguida, repita com os conectores vermelhos. Observa-se essa configuração na figura 37. Una o conector vermelho do amperímetro ao conector vermelho da fonte. Em seguida, una o conector preto do amperímetro a qualquer conector preto da carga resistiva. Por fim, conecte qualquer conector vermelho da carga ao conector A1 do motor, finalizando com a interligação do conector A2 do motor ao conector de referência da fonte. No final, todos os componentes estão conectados e prontos para a simulação. Na figura 38 observam-se todas essas conexões.

Com todos os componentes devidamente conectados é necessário energiza-los. Para tanto, basta acionar o disjuntor geral e os disjuntores das fontes. Seguindo os passos da prática proposta, ajusta-se a carga resistiva em  $60\Omega$  e o reostato em  $10\Omega$ . Nesse momento, aumenta-se a tensão fornecida pela fonte até que o motor alcance uma velocidade de 1500rpm e se observe um valor de 94V. Essa é a primeira forma de controle da velocidade do motor.

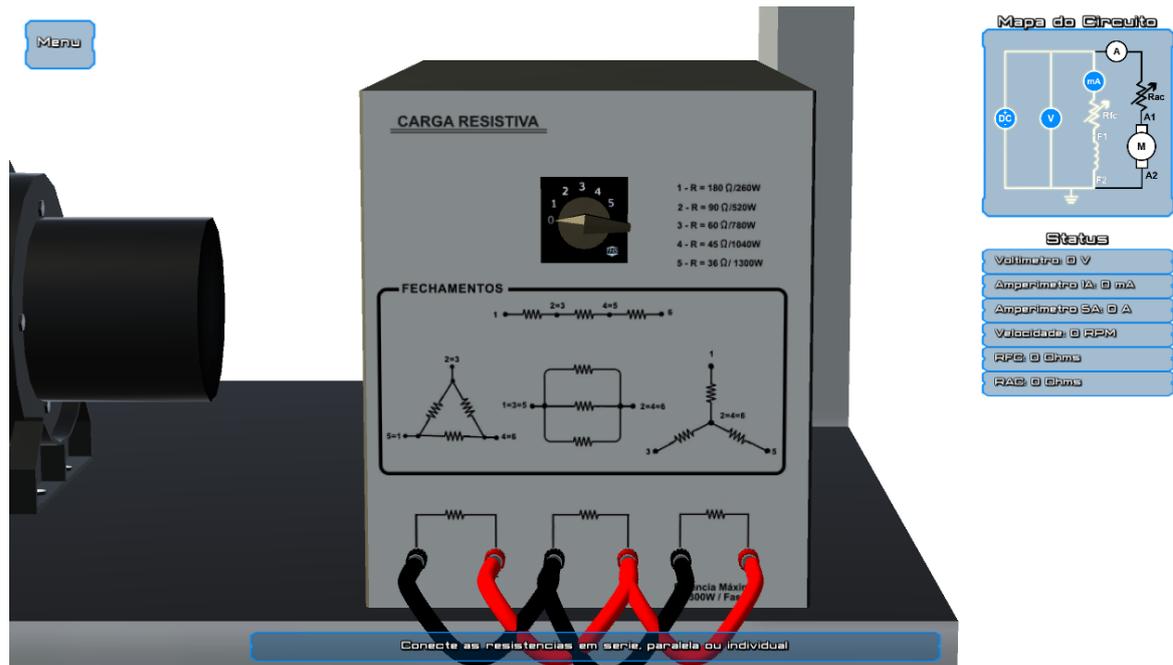


Figura 37 – Simulação da conexão em paralelo da carga resistiva.

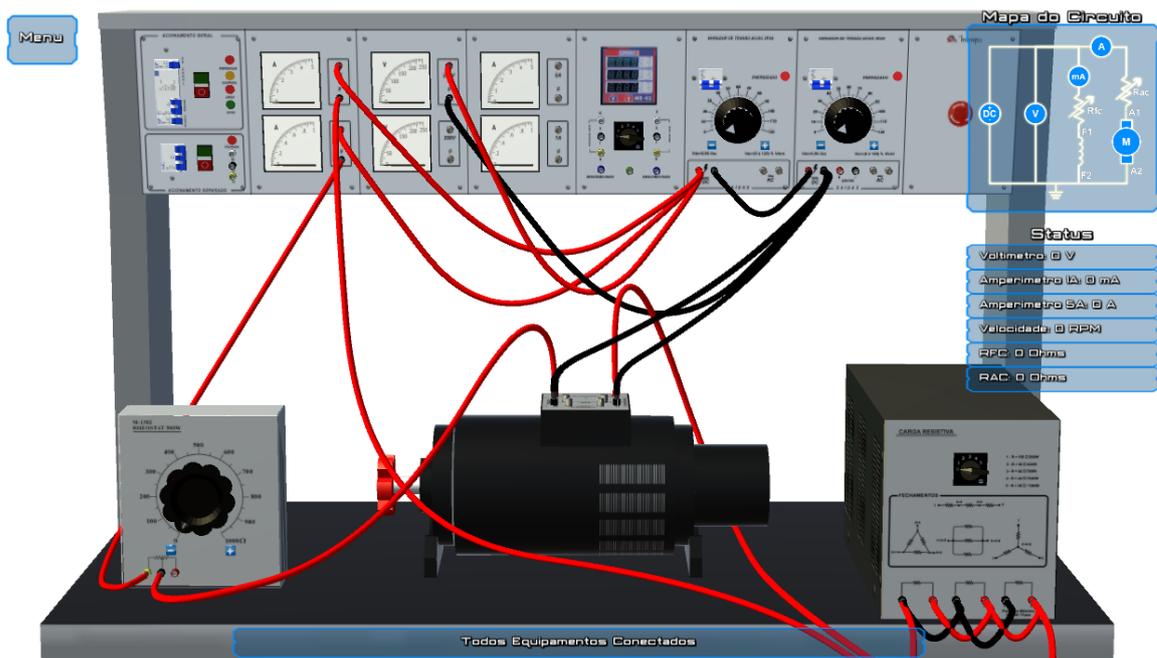


Figura 38 – Simulação com todos os equipamentos conectados.

Tabela 3 – Velocidades do motor pela resistência da armadura

Resistência	Rotação
30Ω	1699rpm
20Ω	1763rpm
15Ω	1796rpm
12Ω	1815rpm

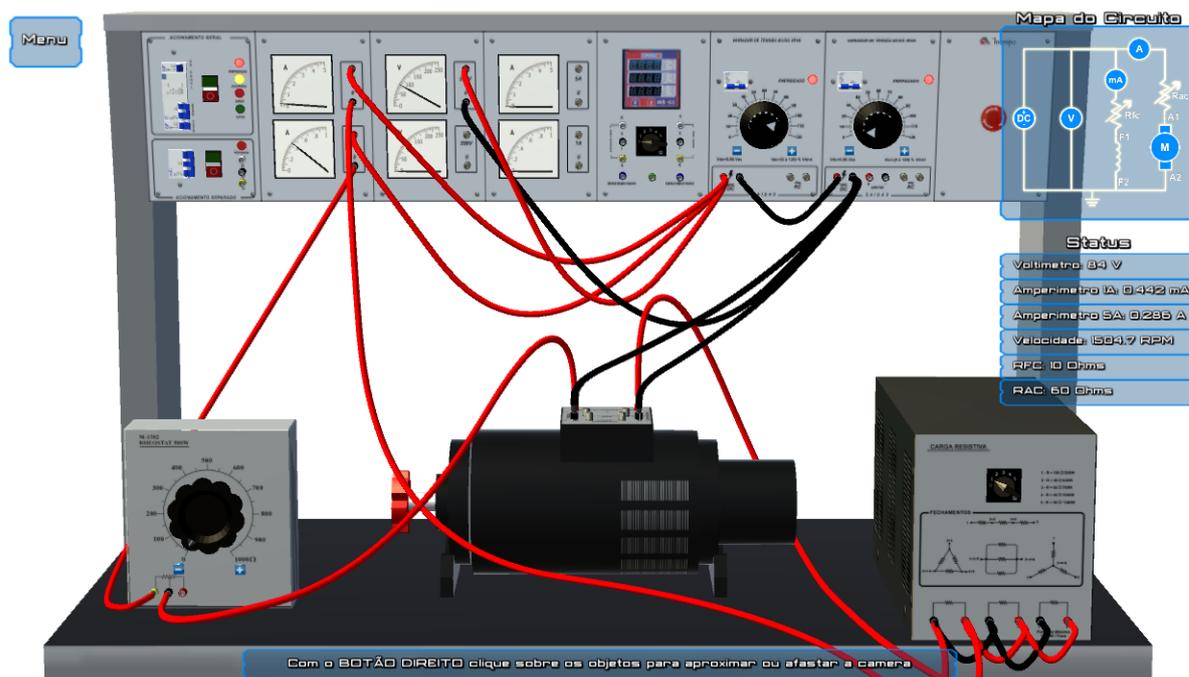


Figura 39 – Simulador em funcionamento

Para um controle de velocidade através da resistência da armadura, basta alterar os valores da carga resistiva entre  $30\Omega$ ,  $20\Omega$ ,  $15\Omega$  e  $12\Omega$  e, então, observar essas velocidades na tabela. Por fim, é preciso retornar ao valor inicial. A última maneira de controlar a velocidade de rotação é por meio do controle do fluxo magnético. Para que isso aconteça, basta alterar o valor do reostato. Em seguida, ajuste o valor até que o motor atinja à velocidade de 1800rpm. Essa velocidade foi obtida com o valor de  $50\Omega$ . Pode-se verificar o funcionamento do simulador na figura 39.

Tabela 4 – Comparação entre os valores obtidos pelo simulador com os valores obtidos no laboratórios

	<b>Simulado</b>	<b>Laboratório</b>
Tensão para 1500rpm	94V	114V
$30\Omega$	1699rpm	1509rpm
$20\Omega$	1763rpm	1630rpm
$15\Omega$	1796rpm	1696rpm
$12\Omega$	1815rpm	1734rpm
Resistência para 1800rpm	$50\Omega$	$69\Omega$

Após os resultados dos testes é possível compará-los aos valores obtidos em laboratório. Essa comparação pode ser observada na tabela 4, quando se observa que os valores obtidos na simulação estão próximos aos valores reais. A diferença encontrada é devido às perdas nos materiais dos equipamentos reais: como a carga resistiva e reostato que não fornecem valores exatos devido ao material, além da variação das propriedades do motor, e a leitura aproximada do equipamento(voltímetro analógico). Nos cálculos da

simulação são utilizados valores ideais, aproximados de valores reais.

### 5.3 Considerações Finais

Este capítulo apresenta os resultados obtidos no desenvolvimento da pesquisa. Nele são abordados os aspectos visuais alcançados durante seu desenvolvimento, bem como, os aspectos de interação com o usuário. Por fim, ele aborda os resultados numéricos atingidos através da interação do usuário com o simulador, comparando-os aos valores atingidos numa prática realizada em laboratório.

Na seção 5.1 duas interfaces para auxílio para o usuário são apresentadas. Na seção 5.2 o funcionamento do simulador é apresentado. Também são realizados testes com a finalidade de comparar os valores simulados com os valores reais. Ao final, foi verificado que os valores são compatíveis e aceitáveis.

## 6 Conclusão

Ao longo deste trabalho é apresentado o desenvolvimento de um simulador para atividades práticas na disciplina de Máquinas Elétricas. Foi proposto um ambiente virtual tridimensional no qual o usuário seja capaz de interagir com os equipamentos encontrados em laboratório, possibilitando uma maior familiaridade com os mesmos. Uma das consequências é que menos erros são cometidos durante a montagem das práticas em laboratório. É perseguindo esse objetivo que este trabalho utiliza técnicas de computação gráfica como, iluminação, texturas, câmera, dentre outras, em conjunto com softwares de modelagem tridimensional e o motor de jogos.

Foi ressaltado ao longo desta pesquisa que dentre as diversas atividades práticas desenvolvidas na disciplina de Máquinas Elétricas escolheu-se abordar, aqui, o desenvolvimento das atividades práticas que consistem em controlar a velocidade de rotação de um motor de corrente contínua.

Com esta pesquisa foi possível constatar que os componentes simulados apresentam características visuais semelhantes aos encontrados no laboratório real. Isso quer dizer que os valores obtidos pelos testes, se comparados aos obtidos em laboratório, apresentam valores aceitáveis. A diferença encontrada é devido às perdas com os materiais dos equipamentos reais.

Dentre as principais dificuldades enfrentadas durante o desenvolvimento deste trabalho podemos destacar a representação dos cabos. Isso devido à natureza complexa do modelo, pois cada cabo apresenta uma forma própria. Outra dificuldade encontrada foram os cálculos da simulação, isso decorre da pouca familiaridade com a disciplina, já que ela não é ofertada no curso de Engenharia de Computação. A interface de interação com os usuários, devido ao número de cenários e de possibilidades possíveis, também apresentou obstáculos.

Como contribuição a trabalhos futuros, novas atividades práticas poderão ser anexadas a esta pesquisa, aumentando, assim, a biblioteca disponível de simulação. Alguns polimentos e melhorias poderão ser realizados nos polígonos e texturas dos objetos, com a finalidade de aumentar o realismo e proporcionar uma melhor interação dos usuários.

## Referências

- 3D MAX, A. *Autodesk 3D MAX - Recursos*. 2016. [Online; accessed 28-Julho-2016]. Disponível em: <<http://www.autodesk.com.br/products/3ds-max/features/all>>. Citado na página 24.
- ALVES, W. P. *Blender 2.63 - Modelagem e Animação*. [S.l.]: Érica, 2012. Citado 2 vezes nas páginas 24 e 32.
- ARAÚJO, A. R. dos S. Agile game process–metodologia ágil para projetos de advergames. *Trabalho de conclusão de graduação em Ciências da Computação, Universidade Federal de Pernambuco*, 2006. Citado 3 vezes nas páginas 17, 18 e 19.
- AZEVEDO, A. C. E. *Computação Gráfica: Geração de imagens*. 14. ed. [S.l.]: CAMPUS - GRUPO ELSEVIER, 2003. Citado 8 vezes nas páginas 8, 19, 20, 21, 22, 23, 40 e 43.
- BARILLI, E.; EBECKEN, N. F. F.; CUNHA, G. G. A tecnologia de realidade virtual como recurso para formação em saúde pública à distância: uma aplicação para a aprendizagem dos procedimentos antropométricos. *Ciênc Saúde Coletiva*, v. 16, n. Suppl 1, p. 1247–56, 2011. Citado 2 vezes nas páginas 14 e 15.
- BARROS, R. L. B. de. Análise de metodologias de desenvolvimento de software aplicadas ao desenvolvimento de jogos eletrônicos. *Trabalho de Graduação–Departamento de Informática–Universidade Federal de Pernambuco, Recife*, 2007. Citado 3 vezes nas páginas 17, 18 e 19.
- BLACKMAN, S.; WANG, J. *Unity for Absolute Beginners*. [S.l.]: Apress, 2014. Citado na página 25.
- CARVALHO, G. H. P. de. Um modelo preditivo para desenvolvimento de jogos de computador. *Trabalho de Graduação–Departamento de Informática–Universidade Federal de Pernambuco, Recife*, 2006. Citado na página 19.
- CHAPMAN, S. J. *Fundamentos de máquinas elétricas*. [S.l.]: AMGH Editora, 2013. Citado 3 vezes nas páginas 8, 26 e 27.
- CLUA, E. W. G.; BITTENCOURT, J. R. Desenvolvimento de jogos 3d: concepção, design e programação. In: *XXIV Jornadas de Atualização em Informática (JAI) Part of XXIV Congresso da Sociedade Brasileira de Computação*. [S.l.: s.n.], 2005. p. 22–29. Citado 2 vezes nas páginas 23 e 25.
- COHEN, I. H. M. M. *OpenGL - Uma Abordagem Prática e Objetiva*. [S.l.]: Novatec, 2006. Citado 5 vezes nas páginas 8, 19, 20, 21 e 22.
- FALCÃO, E. de L.; MACHADO, L. S. Um laboratório virtual tridimensional e interativo para suporte ao ensino de física. In: *Anais do Workshop de Informática na Escola*. [S.l.: s.n.], 2010. v. 1, n. 1, p. 1285–1293. Citado 2 vezes nas páginas 30 e 31.
- FEIJO B., P. P. C. E. Visualização, simulação e games. *Anais da XXIV Jornada de Atualização em Informática do Congresso da Sociedade Brasileira de Computação*, n. 1, p. 1313–1356, 2006. Citado 3 vezes nas páginas 14, 17 e 25.

- FITZGERALD, A.; KINGSLEY, C.; UMANS, S. *Máquinas Elétricas - 6.ed.: Com Introdução à Eletrônica de Potência*. [S.l.]: BOOKMAN COMPANHIA, 2006. Citado 3 vezes nas páginas 26, 27 e 28.
- FLAVELL, L. *Beginning Blender: Open Source 3D Modeling, Animation, and Game Design*. [S.l.]: Apress, 2010. Citado 2 vezes nas páginas 24 e 41.
- GAMASUTRA. *Gamasutra salary survey 2014*. 2014. [Online; accessed 16-Julho-2016]. Disponível em: <<http://www.gamasutra.com/salariesurvey2014.pdf>>. Citado na página 18.
- GIBSON, J. *Introduction to Game Design, Prototyping, and Development: From Concept to Playable Game with Unity and C#*. [S.l.]: Addison-Wesley Professional, 2014. Citado 3 vezes nas páginas 25, 32 e 34.
- HEARN M. PAULINE BAKER, W. R. C. D. *Computer Graphics with OpenGL*. 4. ed. [S.l.]: Prentice Hall, 2011. Citado 4 vezes nas páginas 19, 20, 21 e 23.
- JUNIOR, A. B. A.; CARDOSO, A.; JR, E. L. Estudo de máquinas elétricas auxiliado por técnicas de realidade virtual. *Anais do WRVA*, v. 8, p. 85–91, 2008. Citado 3 vezes nas páginas 14, 15 e 30.
- LIMONGELLI, C. et al. Virtual industrial training: Joining innovative interfaces with plant modeling. In: *Information Technology Based Higher Education and Training (ITHET), 2012 International Conference on*. [S.l.: s.n.], 2012. p. 1–6. Citado 2 vezes nas páginas 14 e 15.
- MAMONE, M. Introducing development tools and monodevelop. *Practical Mono*, Springer, p. 21–40, 2006. Citado na página 34.
- MARTIGNONI, A. *Máquinas Elétricas de Corrente Contínua*. [S.l.]: Globo, 1971. Citado 2 vezes nas páginas 26 e 27.
- MAYA, A. *Autodesk Maya - Recursos*. 2016. [Online; accessed 28-Julho-2016]. Disponível em: <<http://www.autodesk.com.br/products/maya/features/all>>. Citado na página 24.
- MOJO, B. O. *STUDIO MARKET SHARE - 2015*. 2016. [Online; accessed 28-Julho-2016]. Disponível em: <<http://www.boxofficemojo.com/studio/?view=company&view2=yearly&yr=2015&debug=0&p=.htm>>. Citado na página 14.
- NEWZOO. *The global games market reaches \$99,6 Billion in 2016, mobile generating 37%*. 2016. [Online; accessed 28-Julho-2016]. Disponível em: <<https://newzoo.com/insights/articles/global-games-market-reaches-99-6-billion-2016-mobile-generating-37/>>. Citado na página 14.
- PHARR, M.; HUMPHREYS, G. *Physically Based Rendering, Second Edition: From Theory To Implementation*. [S.l.]: Morgan Kaufmann, 2010. Citado 2 vezes nas páginas 20 e 21.
- ROCHA, D. C. et al. Desenvolvimento de uma plataforma auxiliar para ensino de máquinas elétricas empregando realidade virtual. In: . [S.l.]: Comissão Especial de Informática na Educação, 2013. Citado 2 vezes nas páginas 14 e 15.

SLOPER, T. Following up after the game is released: It's not over when it's over. *Game Design Perspectives*, Charles River Media, p. 327, 2002. Citado 2 vezes nas páginas 18 e 19.

SOUZA, A. L. de et al. Ambiente multimídia para o ensino de máquinas elétricas: Uma abordagem multidisciplinar. *Revista de Ensino de Engenharia*, v. 30, n. 1, p. 45–51, 2012. Citado na página 29.

TORI CLAUDIO KIRNER, R. S. R. *Fundamentos e Tecnologia de Realidade Virtual e Aumentada*. [S.l.]: SBC, 2006. Citado 3 vezes nas páginas 17, 23 e 24.

# Apêndices

## Modelos dos Objetos



Figura 40 – Objeto 3D referente a bancada didática



Figura 41 – Objeto 3D referente ao motor



Figura 42 – Objeto 3D referente ao reostato



Figura 43 – Objeto 3D referente da carga resistiva

## Fotos dos objetos reais

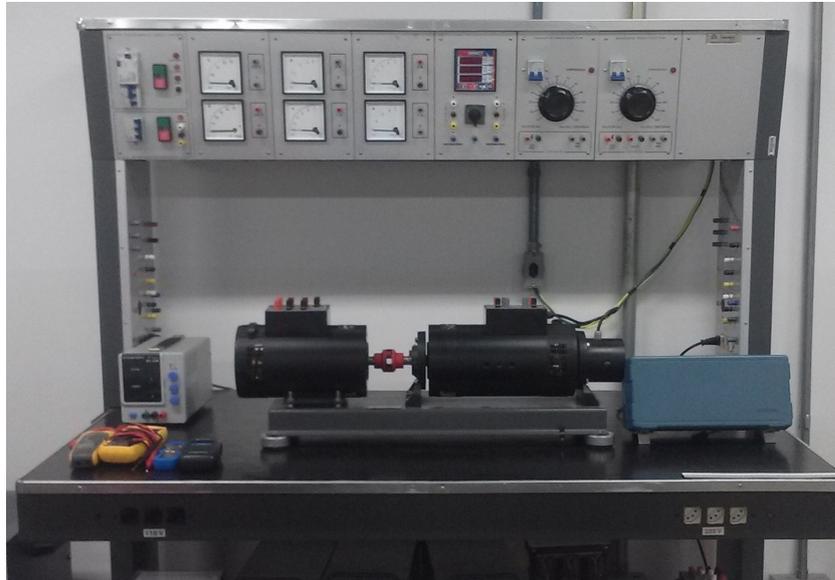


Figura 44 – Foto da Bancada Didática



Figura 45 – Foto do Reostato



Figura 46 – Foto do Motor

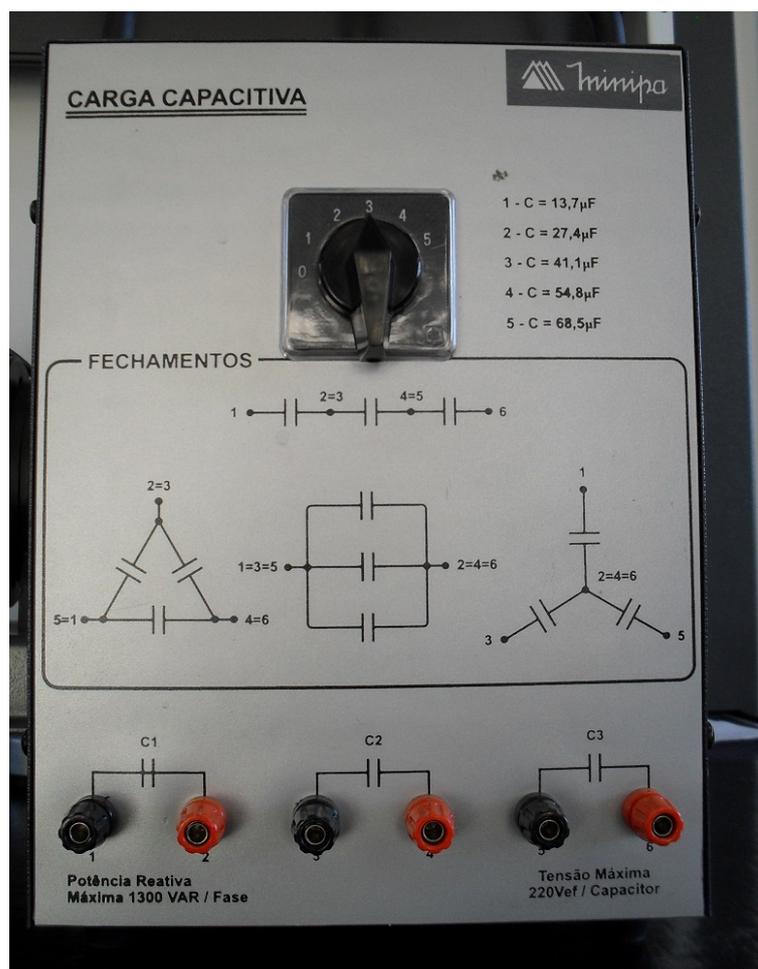


Figura 47 – Foto da Carga Resistiva