

**UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANÁ  
DIRETORIA DE PESQUISA E PÓS-GRADUAÇÃO  
PROGRAMA DE PÓS-GRADUAÇÃO EM TECNOLOGIAS COMPUTACIONAIS  
PARA O AGRONEGÓCIO**

**ALEXSSANDER ARIEL KUHN LIESENFELD**

**PROGRAMAÇÃO INTEGRADA APLICADA À AUTOMAÇÃO DE  
PROJETOS: UMA APLICAÇÃO EM EXAUSTORES PARA AVIÁRIOS**

**DISSERTAÇÃO**

**MEDIANEIRA**

**2017**

ALEXSSANDER ARIEL KUHN LIESENFELD

**PROGRAMAÇÃO INTEGRADA APLICADA À AUTOMAÇÃO DE  
PROJETOS: UMA APLICAÇÃO EM EXAUSTORES PARA AVIÁRIOS**

Dissertação apresentada ao Programa de Pós-Graduação em Tecnologias Computacionais para o Agronegócio da Universidade Tecnológica Federal do Paraná – UTFPR – Campus Medianeira como requisito parcial para obtenção do grau de Mestre em Tecnologias Computacionais para o Agronegócio.

Orientador: Prof. Dr. Paulo César Tonin.

**MEDIANEIRA**

**2017**

À minha namorada, Lissandra Mazurana,  
pelo apoio, compreensão, incentivo e carinho  
em todos os momentos.

Aos meus pais, Marcelo e Clarice, que me  
presentearam com a educação e são meus  
exemplos e minha força nos obstáculos mais  
difíceis.

## **AGRADECIMENTOS**

Agradeço primeiramente à Deus, pela iluminação e inspiração constante.

Aos meus familiares, meu irmão Alysson, meus tios e meus avós, e todos que de uma forma ou de outra participaram deste desafio e me ajudaram a concluí-lo, seja com palavras, com gestos, com apoio em momentos difíceis ou com um simples “parabéns” a cada passo concluído. A eles minha eterna gratidão e reconhecimento.

Reverencio ao meu professor orientador Paulo César Tonin, pela sua enorme prestatividade e condução exímia durante as pesquisas e execução deste trabalho. Agradeço ao professor Claudio Leones Bazzi, por ter acreditado em mim e me apoiado em todos os momentos em que precisei. Ofereço meus cumprimentos, também, a cada professor com os quais tive o privilégio de ter contato no decorrer desta jornada e receber todo o conhecimento que se propuseram, constantemente, em nos oferecer.

À Universidade Tecnológica Federal do Paraná, por disponibilizar o local e permitir que este projeto esteja se concluindo.

Enfim, sou grato a todas as pessoas que de qualquer maneira contribuíram com o sucesso deste projeto. À todos estes, meus sinceros e eternos agradecimentos.

*“A persistência é o caminho do êxito”.*

(Charles Chaplin)

*“O heroísmo não consiste em não ter medo, mas sim em superá-lo”.*

(Roberto Gómez Bolaños)

## RESUMO

LIESENFELD, Alexssander Ariel Kuhn. **Programação integrada aplicada à automação de projetos: Uma aplicação em exaustores para aviários**. 2017. 72 folhas. Dissertação (Mestrado em Tecnologias Computacionais para o Agronegócio) – Universidade Tecnológica Federal do Paraná. Medianeira, 2017.

**RESUMO:** Este trabalho trata de uma alternativa na execução do projeto de um exaustor utilizado em aviário, totalmente automatizada. Para isto, utiliza-se a integração de um *software* de CAD 3D (SolidWorks®), através de sua API, com outro *software* responsável por permitir a codificação de macros (Visual Basic for Applications®), as quais sendo executadas geram todo o projeto do equipamento. São utilizadas interfaces de interação com o usuário para que este determine as principais medidas geométricas do equipamento e estes dados sejam enviados à macro, que converte-os de modo que fiquem compatíveis com o necessário para a execução e inicie a modelagem do exaustor. Ao final da execução do código, o usuário terá o projeto completo de um exaustor de acordo com as medidas definidas. O projeto do equipamento ficará salvo no diretório informado pelo usuário, em formato de arquivo padrão do *software* de CAD 3D, assim como os detalhamentos em formato PDF. Desta forma os arquivos podem ser abertos até mesmo em máquinas que não possuem o *software* de CAD 3D. Considerando a importância do trabalho para projetistas e estudantes leigos na programação, publicou-se um livro onde pode-se verificar todos os aspectos apresentados neste trabalho e, além de conseguir desenvolver este mesmo projeto, fazer qualquer outro tipo de automação dentro do *software*. O endereço eletrônico para acesso deste material é: "<https://goo.gl/9IKCuN>".

**PALAVRAS-CHAVE:** Exaustor, SolidWorks® API, Visual Basic for Applications®.

## ABSTRACT

LIESENFELD, Alexssander Ariel Kuhn. **Programação integrada aplicada à automação de projetos: Uma aplicação em exaustores para aviários**. 2017. 72 folhas. Dissertação (Mestrado em Tecnologias Computacionais para o Agronegócio) – Universidade Tecnológica Federal do Paraná. Medianeira, 2017.

**ABSTRACT:** This paper deals with an alternative in the execution of an exhaust-fan project used in aviary, fully automatized. In order to achieve that intention, the integration of a 3D CAD software (SolidWorks®) through its API and another software responsible for allowing the coding of macros is used (Visual Basic for Applications®), and when in performance, it creates the entire purpose of the equipment. User interaction interfaces are used to determine the equipment's main geometric measurements and this data is sent to the macro, converting them so that they are compatible with what is required for the execution and the start of the exhaust-fan modeling. At the end of the coding execution, the user will have the complete design of an exhaust-fan according to the defined measures. The design of the equipment will be saved in the directory entered by the user, in 3D CAD software format, as well as in PDF file format for detailing. This way, files can be opened even on computers/machines that do not have a 3D CAD software. Considering the importance of the work for designers and lay students in programming, a book was published so that one can verify all the aspects presented in this paper and, besides being able to develop this same project, also making any other kind of automation within the software. The electronic address for accessing this material is: "<https://goo.gl/9IKCuN>".

Keywords: Exhaust-fan, SolidWorks API®, Visual Basic for Applications®.

## LISTA DE FIGURAS

Figura 1 - Fluxograma de execução de equipamento na indústria.....	15
Figura 2 - Disposição de tarefas do VBA. ....	17
Figura 3 - Produção Brasileira de Carne de Frango (Milhões Ton.).....	18
Figura 4 - Exportação Brasileira de Carne de Frango (Mil Ton.).....	19
Figura 5 - Percentual de abate de frango por estado brasileiro.....	20
Figura 6 - Estados exportadores de carne de frango em 2015. ....	20
Figura 7 - Peça modelada em 3D no SolidWorks®.....	23
Figura 8 - Detalhamento da peça feito no SolidWorks® .....	24
Figura 9 - Código em linguagem Visual Basic gerado pela gravação de macro. ....	26
Figura 10 - Vista externa do conjunto de exaustores. ....	28
Figura 11 - Sistema de exaustão do aviário. ....	28
Figura 12 - Exaustores analisados para serem utilizados como modelo padrão.....	32
Figura 13 - Polia utilizada como modelo. ....	33
Figura 14 - Grade frontal utilizada como modelo.....	33
Figura 15 - Medição dos componentes. ....	34
Figura 16 - Fluxograma de execução do modelo inicial. ....	35
Figura 17 - Fluxograma de execução das macros e formulários.....	37
Figura 18 - Modelo utilizado como padrão. ....	38
Figura 19 - Principais medidas do modelo. ....	39
Figura 20 - Trecho do código inicial de modelagem da carcaça. ....	50
Figura 21 - Interface inicial da API Help. ....	51
Figura 22 - Arquivos dos códigos que geram as peças.....	52
Figura 23 - Exemplo de codificação de posicionamento. ....	53
Figura 24 - Montagem final do exaustor modelo. ....	54
Figura 25 - Exemplo de codificação dos detalhamentos. ....	55
Figura 26 – Exemplos de detalhamentos gerados pela execução da macro. ....	56
Figura 27 - Declaração das variáveis a serem definidas pelo usuário. ....	57
Figura 28 - Modelos com número de hélices diferente.....	58
Figura 29 - Definição de medida dinamicamente. ....	59
Figura 30 - Mudança dos valores de milímetros para metros.....	59
Figura 31 - Aviso de execução do projeto concluída.....	60
Figura 32 - Trecho de código responsável por delimitar as medidas. ....	60

Figura 33 - Adição de referência ao Microsoft Office® 2013.....	61
Figura 34 - Código de seleção de formato de folha e diretório.....	62
Figura 35 - Árvore de projetos do VBA, depois da codificação finalizada. ....	63
Figura 36 - Formulário de apresentação. ....	64
Figura 37 - Formulário de inserção de dados.....	65

## LISTA DE QUADROS

Quadro 1 - Peças desenvolvidas para exaustor modelo.....	40
Quadro 2 - Montagens desenvolvidas para exaustor modelo .....	43
Quadro 3 - Esquema das folhas de detalhamento desenvolvidas para o exaustor modelo .....	45

## LISTA DE ABREVIATURAS E SIGLAS

2D	2 dimensões
3D	3 Dimensões
ABNT	Associação Brasileira de Normas Técnicas
ABPA	Associação Brasileira de Proteína Animal
AGEITEC	Agência Embrapa de Informação Tecnológica
API	Application Programming Interface
BASIC	Beginners All-Purpose Symbolic Instruction Code
CAD	Computer-Aided Design
CAPES	Coordenação de Aperfeiçoamento de Pessoal de Nível Superior
GPS	Global Positioning System
GUI	Graphical User Interface
OLE	Object Linking and Embedding
PDF	Portable Document Format
UTFPR	Universidade Tecnológica Federal do Paraná
VB	Visual Basic
VBA	Visual Basic for Applications

## SUMÁRIO

1	INTRODUÇÃO .....	12
1.1	OBJETIVO GERAL .....	13
1.2	OBJETIVOS ESPECÍFICOS .....	13
2	JUSTIFICATIVA .....	14
3	FUNDAMENTAÇÃO TEÓRICA.....	18
3.1	O CENÁRIO DA AVICULTURA .....	18
3.2	INDÚSTRIA E EQUIPAMENTOS PARA AVICULTURA .....	21
3.3	IMPORTÂNCIA DO PROJETO PARA FABRICAÇÃO DE EQUIPAMENTOS ....	22
3.4	O SOLIDWORKS® E SUA UTILIZAÇÃO .....	23
3.5	USO DE FERRAMENTA API DO SOLIDWORKS® .....	25
3.6	APLICAÇÃO DA API NA INDÚSTRIA .....	27
4	MATERIAIS E MÉTODO.....	30
4.1	PROCEDIMENTOS METODOLÓGICOS .....	30
4.2	LEVANTAMENTO DE CONTEÚDO COM ENFOQUE NA ÁREA.....	31
4.3	DEFINIÇÃO DE MODELO PADRÃO E MODELAGEM DO EXAUSTOR .....	31
4.4	CRIAÇÃO E EDIÇÃO DOS CÓDIGOS PARA AUTOMAÇÃO .....	35
5	RESULTADOS E DISCUSSÕES.....	38
5.1	COLETA DE INFORMAÇÕES E DEFINIÇÃO DE MODELO PADRÃO.....	38
5.2	MODELAGEM DO EXAUSTOR MODELO .....	39
5.3	DESENVOLVIMENTO DA CODIFICAÇÃO .....	49
5.3.1	Desenvolvimento dos códigos responsáveis por gerar as peças .....	49
5.3.2	Desenvolvimento dos códigos responsáveis por gerar as montagens .....	52
5.3.3	Desenvolvimento dos códigos responsáveis por gerar os detalhamentos.....	54
5.3.4	Utilização de variáveis na automação dos códigos.....	57
5.3.5	Criação dos formulários de interação com o usuário .....	64
6	CONCLUSÕES E TRABALHOS FUTUROS.....	66
6.1	CONCLUSÕES.....	66
6.2	TRABALHOS FUTUROS .....	67
	REFERÊNCIAS BIBLIOGRÁFICAS .....	68

## 1 INTRODUÇÃO

Com o passar dos anos o uso da tecnologia em todas as áreas do conhecimento tem aumentado substancialmente. Seja, por exemplo, no comércio com o cadastro de mercadorias, nas empresas utilizando banco de dados para cadastro de clientes e nas várias utilizações de sistemas de GPS (*Global Positioning System*). Um grande avanço na realização de todo tipo de projeto foi a criação dos *softwares* de desenho auxiliado por computador ou CAD (*Computer-Aided Design*). Estes *softwares* aceleraram o processo de criação de desenhos e projetos que, antes da década de 90, eram feitos a mão e com o uso de muitos profissionais. Com as indústrias não foi diferente, com a utilização destas ferramentas, ficou muito mais fácil desenvolver projetos de peças e equipamentos antes de serem manufaturados.

Na indústria agrícola existem várias empresas que fabricam peças para serem utilizadas no agronegócio. Na avicultura, existem indústrias metal-mecânicas que fabricam exaustores, os quais possuem diversos modelos e tamanhos. A execução de projetos para a fabricação desses equipamentos facilita muito, já que estes possuem várias peças em sua montagem, os quais podem ser mais facilmente manufaturados quando se dispõe um projeto detalhado destas peças. No entanto, para que sejam executados estes detalhamentos, se faz necessário que as empresas possuam projetistas, o que em muitas vezes se torna inviável para pequenas e médias empresas em função do custo desses profissionais. Uma solução que pode ser proposta é a automação dos procedimentos repetitivos no projeto destes equipamentos, a fim de facilitar a execução destas atividades. Para isto, podem ser utilizadas ferramentas de programação vinculada, que faz com que um *software* que permita realizar alguma programação possa ser vinculado a um segundo *software*, que neste caso será um *software* de CAD 3D, além de criar formulários ou interfaces que façam a interação com o usuário e permitam que este defina valores para que o exaustor seja criado automaticamente, fazendo todo o processo repetitivo e alterando os valores que sejam necessários alterar.

Com a criação destas interfaces, uma pessoa, mesmo sem possuir grandes conhecimentos no *software* de CAD 3D, é capaz de realizar a modelagem e detalhamento de um exaustor e de todas as peças que o compõe.

Por se tratar de um assunto com grande abrangência na engenharia, este tipo de trabalho torna-se de grande relevância para empresas, indústrias, projetistas e acadêmicos de diversas áreas, pois trabalhos com ferramentas de CAD são amplamente utilizados em diferentes áreas do conhecimento. Da mesma forma como este trabalho irá abordar a automação na elaboração de projetos de exaustores para aviários, estas ferramentas podem ser utilizadas para automatizar os diversos processos de criação de projetos na indústria bem como ser utilizada para realizar simulações com diferentes tamanhos de peças e equipamentos, dentre várias outras aplicações que podem ser realizadas.

## 1.1 OBJETIVO GERAL

O trabalho tem por objetivo geral realizar a codificação de macros responsáveis por automatizar a criação do projeto de um exaustor utilizado em aviários e criar interfaces que permitam a interação do usuário com o *software* de CAD 3D, de modo que este entre com os valores referentes às principais medidas geométricas com os quais deseja que o exaustor seja projetado e o *software* realize todo o procedimento de modelagem das peças, montagem e detalhamentos deste equipamento.

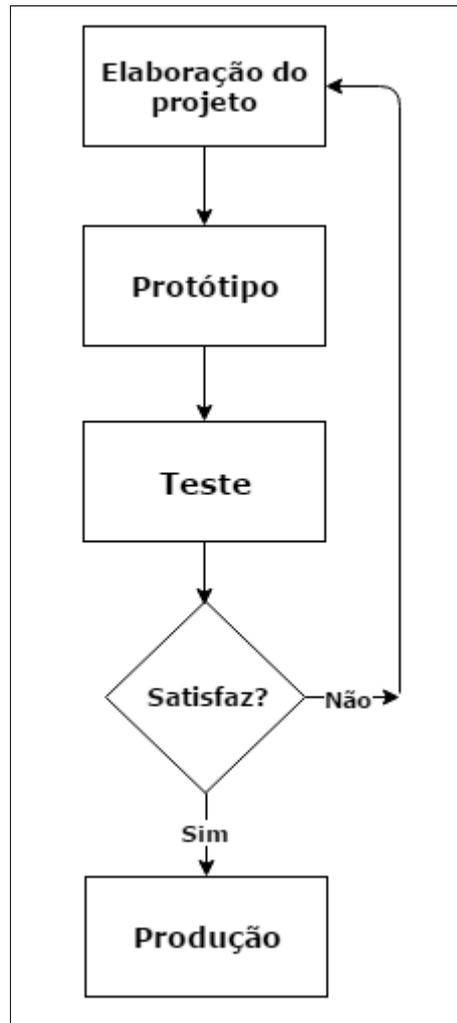
## 1.2 OBJETIVOS ESPECÍFICOS

- a) Obter imagens fotográficas, a campo, dos componentes usados no exaustor;
- b) Realizar a modelagem no *software* de CAD 3D de todos os componentes do exaustor, fazer a montagem e realizar o detalhamento do equipamento;
- c) Desenvolver os códigos responsáveis por automatizar o projeto utilizando a API (*Application Programming Interface*) do *software* e, sempre que necessário, os recursos de gravação e edição de macro;
- d) Realizar os testes e otimizar o código em Visual Basic, de modo que se torne dinâmico e se modifique de acordo com medidas especificadas pelo usuário;
- e) Desenvolver e validar as interfaces para interação com o usuário.

## 2 JUSTIFICATIVA

O trabalho a ser desenvolvido justifica-se pela importância de se ter um bom projeto para a fabricação de qualquer tipo de equipamento, o que significa diminuir os erros durante a manufatura, economizando tempo e matéria prima. Para que um projeto seja desenvolvido em um *software* CAD é necessário ter na empresa uma pessoa com experiência no *software*, o que muitas vezes não acontece, dificultando a execução deste importante trabalho. Oliveira (2007) relata que, em média, das empresas de pequeno e médio porte que trabalham no setor metal-mecânico no Brasil e utilizam algum tipo de desenho computacional em seus projetos, apenas 50% utiliza projetos em 3D. Isto significa que atualmente existem empresas que trabalham com projetos utilizando desenhos 2D e/ou modelos físicos, de modo que, com os métodos mais eficientes existentes e tecnologias avançadas, torna-se interessante utilizar projetos em 3D e simulação para determinar os componentes e recursos a serem utilizados em sua fabricação e até mesmo sequência de montagem (FARHAN, 2011). Ciclos de desenvolvimento de projetos sem a utilização de softwares de CAD são caros e longos e projetos realizados em CAD 3D permitem reduzir este número de ciclos. Também podem reduzir custos diminuindo erros de fabricação, o que significa menos material gasto e menos retrabalho sendo realizado. Outro fator interessante é que, quando o equipamento é desenvolvido mais rapidamente, aumenta o tempo de exposição e conseqüente venda deste produto. Com a execução de projetos em 3D é possível simular diferentes tamanhos do equipamento a ser desenvolvido antes de se tomar qualquer decisão final, outro fator que influencia diretamente no consumo de tempo e de material, sendo esses, aspectos muito relevantes para qualquer organização.

Na Figura 1, pode-se verificar como deve ser tratada a execução do projeto de um equipamento até a chegada à etapa de produção do mesmo, seguindo o cronograma no intuito de gastar o mínimo possível de tempo e material.



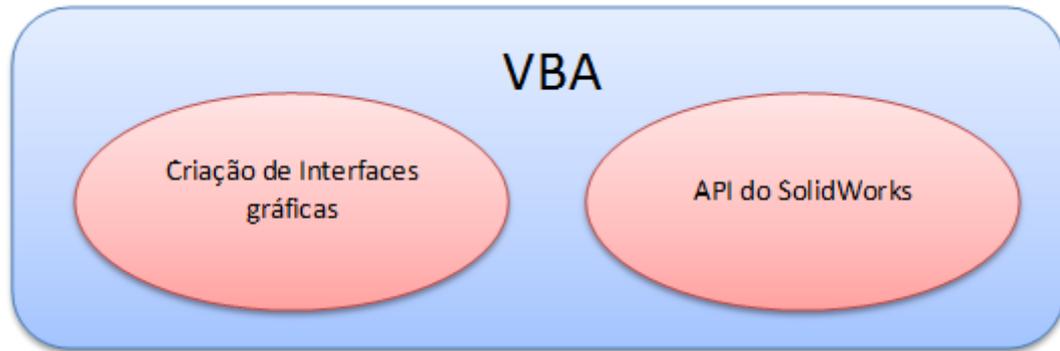
**Figura 1 - Fluxograma de execução de equipamento na indústria.**  
**Fonte: Autoria Própria.**

Devido a estes fatos, torna-se interessante encontrar formas de fazer com que estas empresas de pequeno e médio porte tenham acesso a projetos mais bem elaborados, sem ter que gastar muito, que muitas vezes inviabiliza a execução destes projetos. Automatizar a execução de projetos específicos de uma empresa, de modo que esta não precise ter um projetista especializado trabalhando diariamente é uma alternativa interessante para resolução deste problema. A solução encontrada, portanto, foi encontrar uma forma na qual o projetista e programador crie todo o projeto do equipamento uma única vez e torne automatizado, de maneira que o usuário tenha a possibilidade de apenas ajustar as principais medidas e características que deseja alterar e o programa seja executado criando um novo projeto deste equipamento, mas com as configurações atualmente definidas pelo usuário.

Um *software* muito poderoso na modelagem 3D é o SolidWorks®, ele dispõe de uma ferramenta bastante importante na automação de processos repetitivos que é a criação e edição de macros. As macros geram códigos em linguagens de programação que podem ser editadas pelo programador, a fim de torná-las usuais na realização destes projetos. O *software* 3D que foi utilizado na execução deste trabalho é o SolidWorks® Premium 2013.

O SolidWorks® disponibiliza ao usuário a possibilidade de utilizar a API, que é um conjunto de rotinas e padrões de programação utilizado para ter acesso a partir de um *software* à outro que, no caso, é o SolidWorks (FARHAN; BRIEN; RAD, 2012). O fato do SolidWorks® disponibilizar acesso à sua API permite que outros programadores desenvolvam produtos associados ao SolidWorks®. No caso, está se criando um sistema automatizado para criação de um exaustor para aviários, utilizando os recursos de criação e edição de macros e os métodos e propriedades necessários disponibilizados pela API do SolidWorks®.

No presente trabalho, o *software* utilizado para o desenvolvimento dos códigos das macros é o Visual Basic for Applications® (VBA), o qual possui ferramentas para criação das interfaces que serão utilizadas para interação com o usuário e é a plataforma padrão de edição de macros que o SolidWorks® disponibiliza, apesar de possuir outras plataformas compatíveis. A linguagem a ser utilizada para a programação é a própria linguagem Visual Basic que, de acordo com Prince, Ryan e Mincer (2013), é uma linguagem de programação baseada na linguagem mais antiga BASIC (*Beginners All-Purpose Symbolic Instruction Code*). Já a palavra “Visual” presente no nome se refere à possibilidade de se criar interfaces gráficas de interação com o usuário (GUI). Ao invés de escrever várias linhas de código que descrevam a aparência e a localização exata de cada elemento do formulário, o usuário apenas seleciona e dimensiona objetos gráficos já criados em sua interface, o que reduz significativamente o tempo disposto com a programação (MICROSOFT CORPORATION, 1998). A Figura 2 ilustra a perspectiva de trabalho em que os recursos citados então inseridos, em que o VBA é responsável por permitir a criação das interfaces gráficas internamente e conectá-las às macros que realizam o projeto do exaustor, além de se conectar diretamente com o SolidWorks®, através da API do *software*, e utilizar todas suas propriedades e métodos.



**Figura 2 - Disposição de tarefas do VBA.**  
**Fonte: Autoria própria.**

Dessa forma, será possível desenvolver o projeto inicial do equipamento, neste caso o de um exaustor a ser utilizado como padrão, para então realizar a modelagem particular de cada peça realizando a gravação e alteração do que for necessário no código por meio do VBA e, então, criar as interfaces de interação com o usuário. Desta forma, qualquer pessoa que venha a realizar o projeto detalhado de algum exaustor a ser fabricado irá abrir o SolidWorks® e executar o código referente a criação do projeto do exaustor, isto fará com que seja exibida a ele a interface para definição dos parâmetros e o programa irá realizar o projeto automaticamente, assim, trará também outro grande benefício importante às empresas que é a redução no tempo de projeto (GAO; SANG, 2008). Isto, conseqüentemente, faz com que empresas pequenas que não possuam projetos em suas fabricações de peças e equipamentos, tenham uma excelente alternativa, sendo necessários apenas um projetista e um programador que realize o projeto e sua automação uma única vez.

### 3 FUNDAMENTAÇÃO TEÓRICA

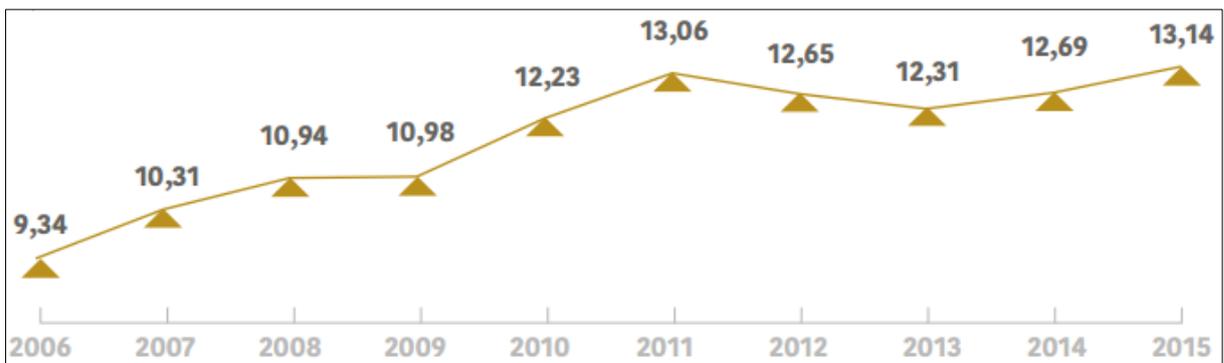
Nesta seção são apresentadas algumas referências de autores que realizaram pesquisas que fundamentam o presente estudo. Também são demonstradas pesquisas realizadas no mesmo âmbito ou com objetivos e atividades semelhantes deste trabalho.

#### 3.1 O CENÁRIO DA AVICULTURA

Há muitos anos o Brasil vem se destacando como uma potência mundial em atividades agropecuárias, principalmente por sua alta produtividade, sendo que a avicultura vem se destacando dentre estas atividades por possuir, principalmente, alta qualidade e baixos custos (CHERNAKI-LEFFER et al., 2002).

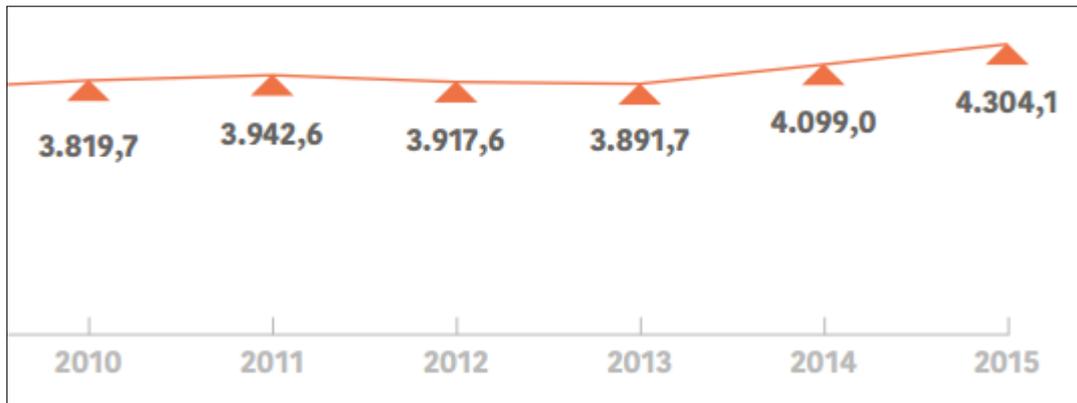
Atualmente o Brasil é o maior exportador e segundo maior produtor de carne de frango no mundo, atrás apenas dos Estados Unidos (ABPA, 2016).

A Associação Brasileira de Proteína Animal (ABPA) calcula que em 2015, a produção brasileira de carne de frango foi de 13,14 milhões de toneladas, de modo que esta foi a maior taxa de produção já registrada no país. O gráfico da Figura 3 mostra os valores da produção de carne de frango no Brasil entre os anos 2006 e 2015. A ABPA informa, ainda, que 2015 foi o ano em que ocorreu o maior número de exportações de carne de frango do Brasil.



**Figura 3 - Produção Brasileira de Carne de Frango (Milhões Ton.).**  
Fonte: Associação Brasileira de Proteína Animal (2016).

A Figura 4 retrata o número de exportações de carne de frango que ocorreram no país entre os anos de 2010 e 2015.



**Figura 4 - Exportação Brasileira de Carne de Frango (Mil Ton.).**  
Fonte: Associação Brasileira de Proteína Animal (2016).

Rodrigues et al. (2014) salientam que a região sul do Brasil é a maior produtora de frango de corte do país, responsável por 57,73% da produção nacional. Os autores também afirmam que o estado do Paraná, desde 2003, se destaca como o maior produtor de carne de frango do país, produzindo em 2012 o equivalente a 3,3 milhões de toneladas, o que representou 26,3% do total nacional produzido naquele ano. O estado também é o que mais abate frangos no Brasil pois, de acordo com a ABPA, o Paraná foi responsável pelo abate, em 2014, de 32,26% do total de abates de frango no país, além de ser o estado que mais exporta o produto, sendo responsável, neste mesmo ano, por 32,21% do total de exportação nacional da carne de frango. A Figura 5 e a Figura 6 mostram, respectivamente, o percentual de abate e de exportação de frango, aos quais cada estado é responsável.

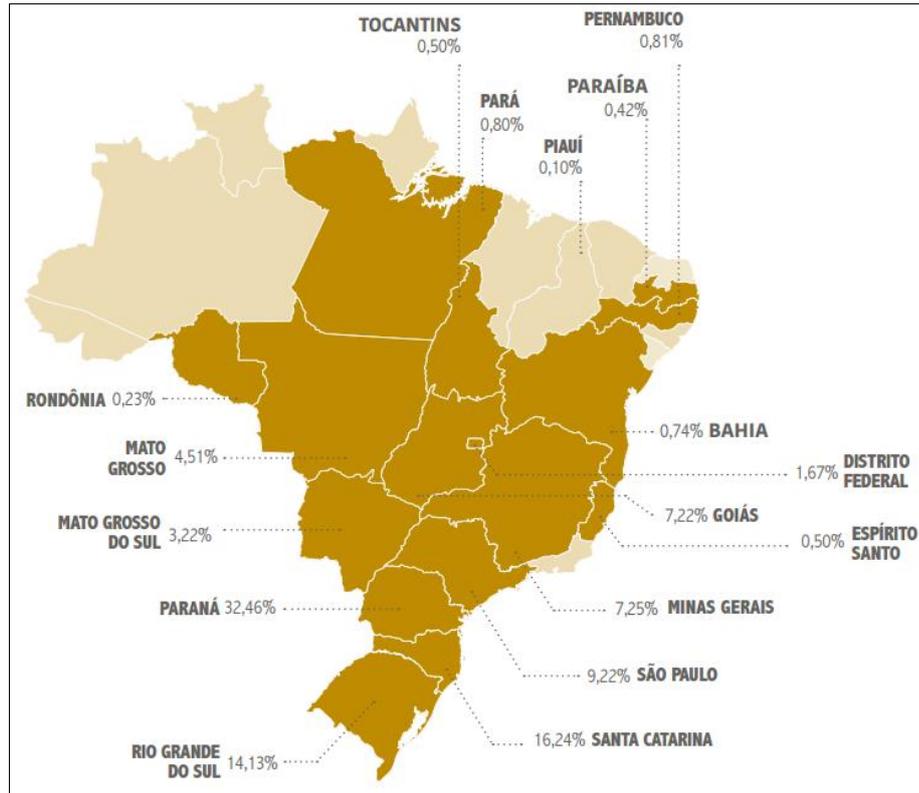


Figura 5 - Percentual de abate de frango por estado brasileiro.  
Fonte: Associação Brasileira de Proteína Animal (2016).

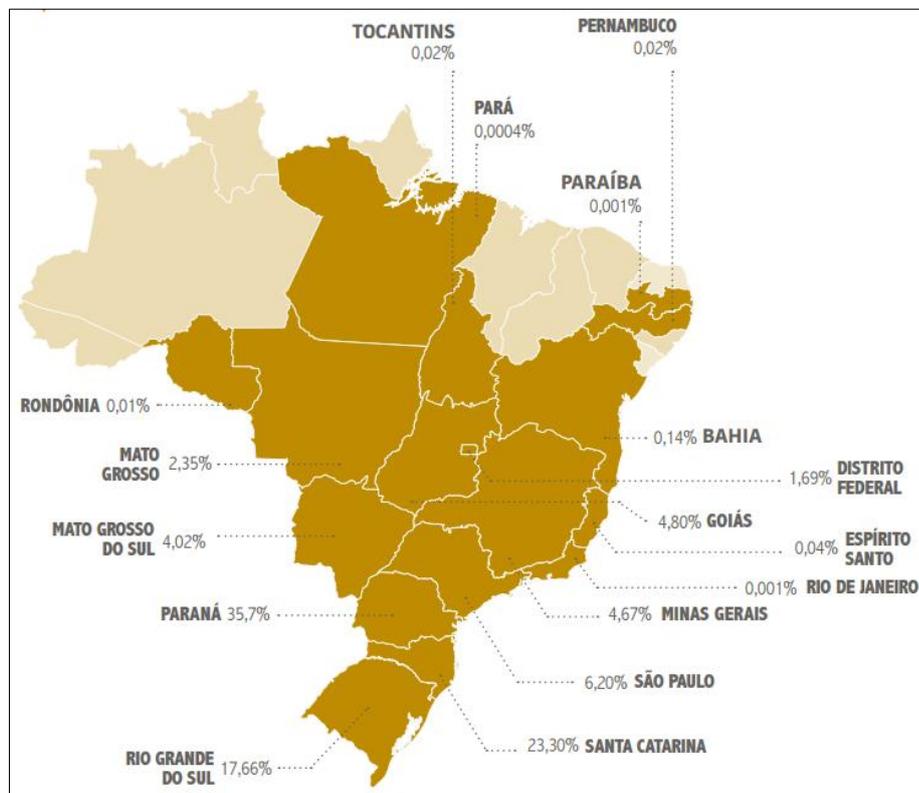


Figura 6 - Estados exportadores de carne de frango em 2015.  
Fonte: Associação Brasileira de Proteína Animal (2016).

Segundo Belusso (2010), a região oeste paranaense é a que mais cresce na produção de aves, onde situam-se as maiores indústrias abatedoras do Brasil. A autora salienta, ainda, que entre os anos de 2000 e 2008, a região oeste paranaense cresceu 112,2% na produção de aves e Chernaki-Leffer et al. (2002) informam que esta região é responsável por 47% da produção de frangos do estado do Paraná.

Desta forma, é possível notar a relevância que tem a agricultura a nível nacional e, mais especificamente a avicultura, tem se mostrado um grande propulsor do PIB nacional até mesmo em épocas em que outras atividades econômicas se mostraram com baixo retorno financeiro. Além disso, de acordo com a Agência Embrapa de Informação Tecnológica (AGEITEC) (2015), estima-se que a cadeia produtiva de frangos de corte era responsável por cerca de 1 milhão de empregos no Brasil em 2012.

Como salientam Cremonez et al. (2015), o fato do Brasil ser atualmente o terceiro maior produtor de frangos de corte, faz com que o processo de produção gere grande quantidade de gordura, cerca de 67,8 gramas por ave, que pode ser utilizada na obtenção de combustível biodiesel.

### 3.2 INDÚSTRIA E EQUIPAMENTOS PARA AVICULTURA

A indústria da produção de frangos vem ano após ano ampliando a escala e incorporando novas tecnologias, desta forma, tornou-se uma atividade que requer constantes atualizações em tecnologia (BELUSSO, 2010). Seguindo este preceito, as indústrias, sejam de pequeno ou médio porte, vêm se modernizando gradativamente, a fim de se tornarem sempre competitivas em um mercado em ascensão como é o da avicultura de corte.

Dimensionamento do espaço interno do aviário, controle de temperatura e ambiência e cuidados com umidade, por exemplo, vêm sendo estudados e aprimorados para melhorar a produtividade dessas aves. Porém, outros fatores que também são muito importantes são os equipamentos utilizados para que os procedimentos de manejo sejam realizados. Podem ser citados vários equipamentos de climatização utilizados em aviários: exaustores, nebulizadores, sistemas de aquecimento, que podem ser a lenha, gás ou até mesmo a diesel, painéis de controle,

onde são programadas a temperatura e a umidade do ar para cada idade da ave, além de cortinas e forração (FUNCK; FONSECA, 2008). Já para o tratamento da 'cama do aviário', Pode-se citar os espalhadores de cal, batedores ou picadores de cama e queimadores de pena, todos os quais podem ser manuais ou em sistemas acoplados ao trator. Pode-se mencionar também, os equipamentos para utilização efetiva dos animais, como os bebedouros (*nipples*) e comedouros entre outros equipamentos importantes, tais como os silos para armazenagem da ração.

### 3.3 IMPORTÂNCIA DO PROJETO PARA FABRICAÇÃO DE EQUIPAMENTOS

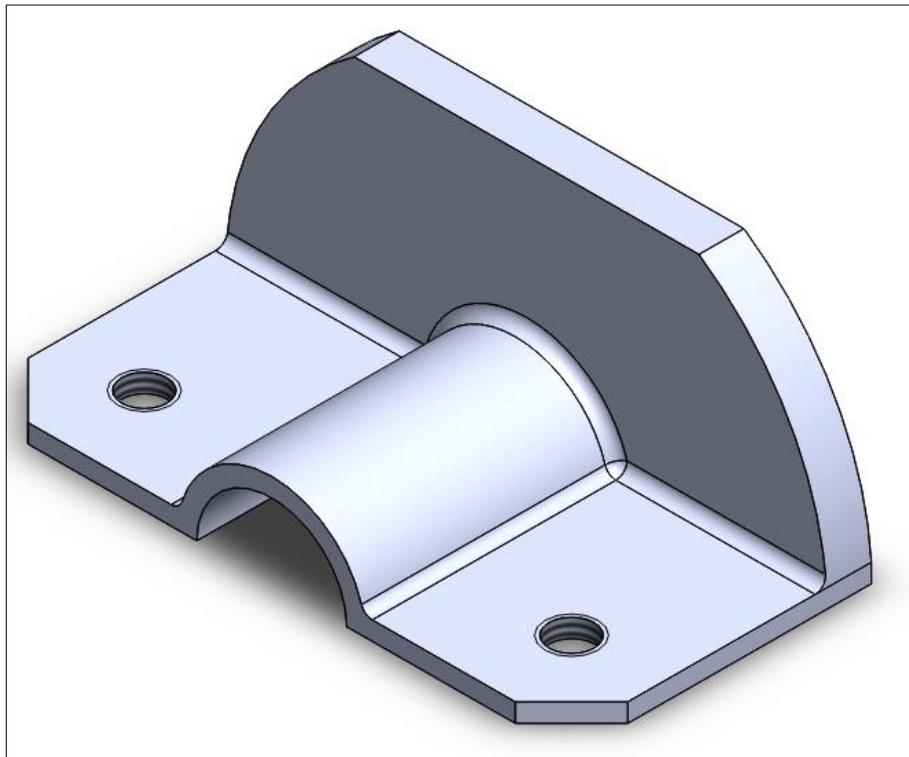
Muitas empresas regionais de pequeno porte, por perceberem o aumento gradativo na demanda por estes equipamentos, começaram a fabricá-los utilizando pouca tecnologia e poucos estudos nos seus desenvolvimentos. Percebe-se, por exemplo, a ausência de qualquer tipo de projeto antes da fabricação efetiva destes equipamentos, sendo que estes procedimentos são úteis para atender as demandas do mercado de forma rápida e eficiente.

Segundo Chakravorty, Chakraborty e Jigeesh (2014), quando se possui um projeto detalhado do equipamento a ser fabricado, o tempo demandado para preparação dos componentes e definição das tarefas é consideravelmente menor. Da mesma forma, de acordo com Ali et al. (2013), utilizando-se um projeto bem elaborado, é possível prever como a peça irá ficar depois de concluída ajudando a detectar possíveis problemas futuros e resolvê-los com antecedência, o que, conseqüentemente, torna o equipamento mais confiável.

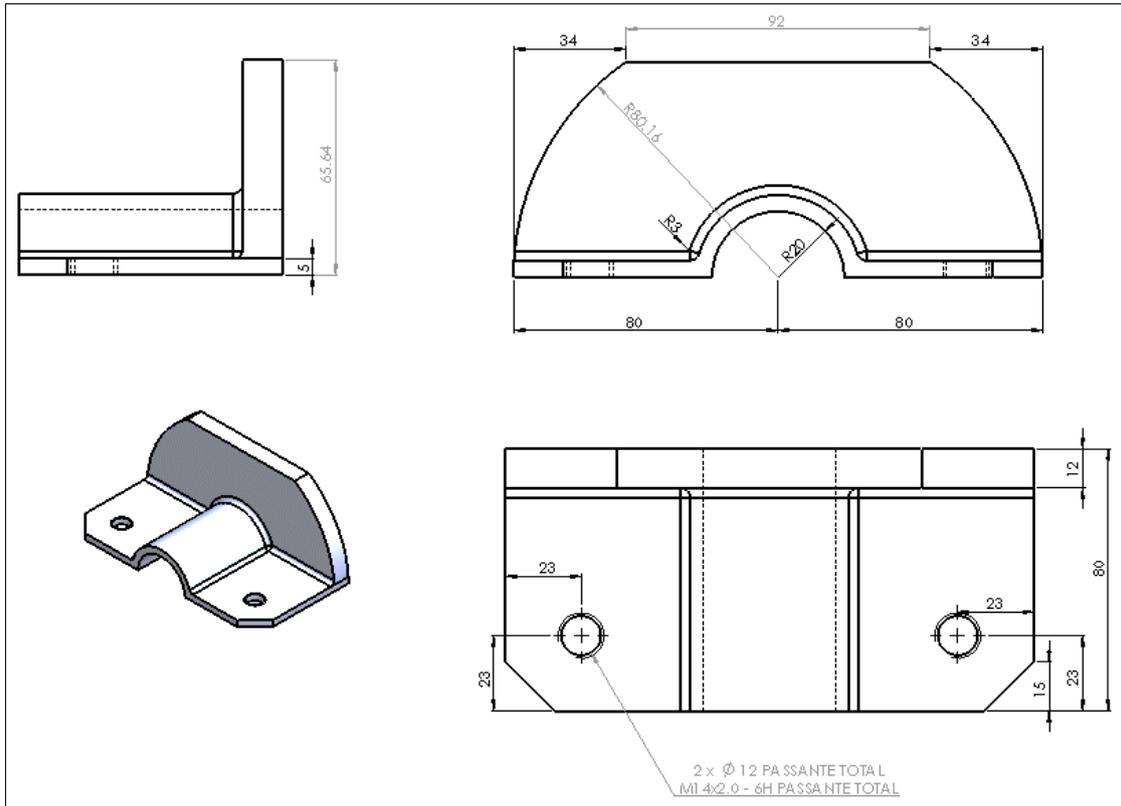
Desenvolver os projetos das peças no computador, antes mesmo de fabricá-las, reduz os custos de produção. Atualmente, vários *softwares* de desenho auxiliado por computador (CAD) foram desenvolvidos com o intuito de realizar este tipo de tarefa e principalmente os *softwares* que trabalham em ambiente 3D, são muito importantes e muito utilizados no projeto de fabricação de qualquer tipo de peça mecânica. O fato do desenho estar em três dimensões, facilita a visualização por completo da peça antes de ser fabricada (ALI et al., 2013).

### 3.4 O SOLIDWORKS® E SUA UTILIZAÇÃO

O SolidWorks® é um *software* de modelagem em ambiente 3D que permite criar e trabalhar com projetos simples e complexos, utilizando ferramentas de desenho computacional que permitem ao usuário modelar peças, realizar a montagem de equipamentos e fazer o detalhamento dessas peças e/ou montagem. Além disto, ele tem outras funcionalidades que possibilitam a simulação estrutural e a simulação de escoamento de fluidos (GAGNEZ, 2013). Devido a essas características, o SolidWorks® tem sido muito usado pela indústria metal-mecânica e faz dele uma excelente ferramenta para o desenvolvimento de projetos dos equipamentos avícolas citados anteriormente. A Figura 7 apresenta uma peça modelada no SolidWorks® 2013 e a Figura 8 apresenta o detalhamento desta mesma peça mostrando as vistas superior, frontal, lateral direita e isométrica.



**Figura 7 - Peça modelada em 3D no SolidWorks®**  
Fonte: Tonin (2015).



**Figura 8 - Detalhamento da peça feito no SolidWorks®**  
 Fonte: Adaptado de Tonin (2015).

Na área agrícola o SolidWorks® também já foi utilizado em trabalhos onde se analisou a deformação de materiais orgânicos em caso de queda (CELIK; RENNIE; IBRAHIM, 2011).

Tinoco et al. (2014) realizaram simulações no SolidWorks® a fim de analisar estruturas de pêndulos de café arábica e sua relevância no amadurecimento e causas de queda dessas frutas.

Groener et al. (2015) fizeram simulações de esforços estruturais na construção de estufas em diferentes regiões e climas, utilizando o SolidWorks®.

Assim como estes citados, são diversos os trabalhos realizados utilizando o SolidWorks® na área agrícola, seja para a realização de análises e simulações, ou na modelagem de equipamento a serem fabricados para serem utilizados nas diversas áreas do agronegócio.

### 3.5 USO DE FERRAMENTA API DO SOLIDWORKS®

O SolidWorks® dispõe de uma biblioteca padrão contendo vários elementos de máquinas, como parafusos, porcas, arruelas e rolamentos, no entanto, se o projetista utilizar com frequência outro tipo de elemento, este deve ser criado e editado sempre que houver necessidade. Nestes casos, o mais adequado seria automatizar o processo de modelagem dessas peças padrão, de modo que o projetista possa dispor deste tempo trabalhando nas etapas mais complexas dos projetos sem se preocupar com as atividades repetitivas de modelagem (SUN; QIN; FANG, 2011).

Com o SolidWorks® é possível realizar estas tarefas repetitivas automaticamente através da gravação de macros. Uma macro quando ativada grava todos os procedimentos que o usuário realiza, permitindo que esta macro seja reproduzida posteriormente e repita os mesmos passos que foram gravados (CHOI; MUN; HAN, 2009). Sendo assim, o usuário pode gravar a modelagem da peça que deseja e, depois, refazer esta mesma peça sempre que quiser utilizar a macro que foi gravada.

Spens (2015) salienta que quando uma macro é gravada, automaticamente um código de programação é gerado, o qual pode ser utilizado para realizar muitas tarefas e alterações, sendo que a maior parte das ações realizadas pelo usuário durante a gravação da macro são gravadas neste código de programação. O SolidWorks® também dá a possibilidade de se trabalhar utilizando sua API (*Application Programming Interface*), a qual Titus e Bin (2013) definem como sendo uma interface que permite que algum *software* interaja com outro utilizando uma linguagem de programação. Assim, modelos criados no SolidWorks® podem ser manipulados diretamente através de outros programas, como é o caso do Visual Basic for Applications®. Devido ao fato de o Visual Basic for Applications® dar a possibilidade de criar interfaces gráficas de interação com o usuário, torna-se possível que uma pessoa, com experiência ou não no SolidWorks®, possa realizar a modelagem do exaustor, definindo suas medidas neste formulário para que, então, o programa execute todas as operações necessárias para a realização do projeto. A Figura 9 apresenta um exemplo de código gerado por uma macro, salvo em formato de linguagem Visual Basic 6.



Utilizando estas técnicas de programação vinculada, Gao e Sang (2008) criaram uma interface gráfica no Visual Basic e fizeram a associação da programação com o SolidWorks®, o qual, utilizando as medidas definidas pelo usuário na interface gráfica, realiza a modelagem e detalhamento de uma chapa perfurada.

### 3.6 APLICAÇÃO DA API NA INDÚSTRIA

Existem diversas aplicações que podem ser realizadas utilizando os códigos de programação disponibilizada pelo SolidWorks®. Mathew (2010) utilizou o Visual Basic for Applications® para criar formulários que interajam com o usuário, propondo que entre com determinadas informações de posicionamento em uma montagem.

Kong et al. (2003) utilizaram a programação gerada pela modelagem de determinadas peças através do SolidWorks®, para a fabricação das mesmas utilizando sistemas de injeção de plástico, através *softwares* alternativos.

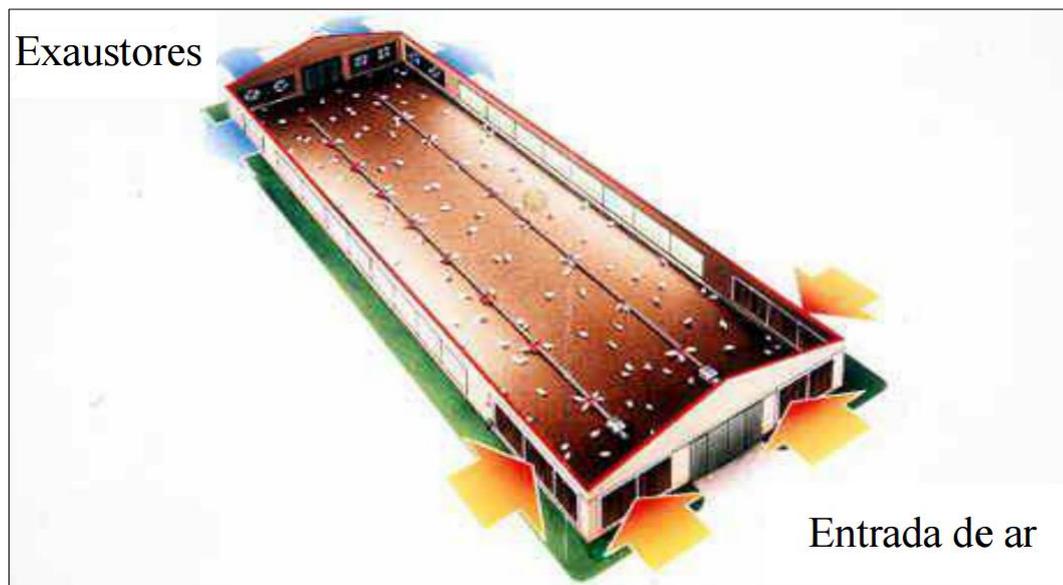
Menda et al. (2014) criaram interfaces utilizando o Visual Basic em que o usuário definia dados correspondentes às peças mecânicas criadas no SolidWorks® e selecionava faces e geometrias as quais eram realizadas simulações de esforços e deformação. Estes dados eram repassados ao *software* Microsoft Excel® utilizando ferramentas de API, e este realizava os referidos cálculos e criava gráficos para demonstrar rapidamente como a peça se comportaria depois de fabricada quando aplicado a ela algum tipo de esforço.

Na indústria avícola, um dos equipamentos fabricados e comercializados são os exaustores, que são responsáveis por retirar o calor do interior do aviário. Bueno e Rossi (2006) salientam que o sucesso ou fracasso na criação de frangos de corte está relacionado diretamente às condições de ambiente às quais são submetidos. A Figura 10 mostra um conjunto de exaustores vistos a partir da área externa do aviário.



**Figura 10 - Vista externa do conjunto de exaustores.**  
**Fonte: Autoria própria.**

A Figura 11 apresenta o processo de exaustão do ar presente no interior do aviário. Devido ao fato do ambiente ser cuidadosamente vedado, o ar é succionado da frente para o fundo, fazendo com que o ar atravesse todo o interior do aviário.



**Figura 11 - Sistema de exaustão do aviário.**  
**Fonte: Carniato et al., 2015.**

O fato de que os exaustores podem ser fabricados com diferentes dimensões ou número de pás, por exemplo, torna interessante para as indústrias fabricantes destes equipamentos, que elas possuam projetos a serem utilizados para auxiliar na manufatura e montagem destas peças. O SolidWorks® sendo uma poderosa ferramenta de modelagem 3D torna-se importante no projeto, e a possibilidade desse

ser vinculado à uma interface criada no Visual Basic for Applications, faz com que seja possível a automação na criação do projeto desses modelos. Esta é a proposta do presente trabalho.

## 4 MATERIAIS E MÉTODO

São apresentados, nesta seção, os procedimentos que foram tomados quando da realização do projeto, sejam eles na realização da pesquisa ou na execução do trabalho propriamente dito.

### 4.1 PROCEDIMENTOS METODOLÓGICOS

O trabalho realizado, que é o processo de modelagem, montagem e detalhamento automatizado dos equipamentos, segue uma abordagem qualitativa por ser uma pesquisa que não pode ser traduzida em números e não necessita técnicas e métodos estatísticos (SILVA; MENEZES, 2005).

Segundo Kauark et al. (2010) esta consiste em uma pesquisa aplicada, pois utiliza de estudos, bases teóricas e até mesmo *softwares* já desenvolvidos na realização do trabalho. Gera conhecimentos para aplicação prática do objeto de pesquisa, buscando resolver problemas específicos na área pesquisada, neste caso, automação na elaboração de projetos.

Pode ser definida por uma pesquisa explicativa (GIL, 2010), podendo, assim, analisar o comportamento do projeto executado com diferentes tipos de medidas dentro de um *software* computacional.

De acordo com Cardoso (2011), também se configura como pesquisa operacional, pois é uma ciência aplicada com o objetivo de melhoria de atividades, neste caso, industriais, realizada utilizando o auxílio de ferramentas computacionais.

O foco principal do trabalho é desenvolver uma interface que interage com o usuário e que permite a este determinar os dados que devem ser utilizados quando da criação do projeto. Desta forma, o trabalho aborda estudos e pesquisas desde a área da computação gráfica, pela utilização do *software* de CAD 3D SolidWorks®, linguagem de programação, neste caso a linguagem Visual Basic 6 especificamente, além do desenvolvimento de interfaces gráficas através do Visual Basic for Applications®.

Pode-se afirmar, também, que foi desenvolvido um novo *software*, levando em conta a definição de MSD (1999) que explica *software* como sendo um conjunto de instruções e regras que deverão ser executadas pelo computador com o objetivo de resolver problemas e agilizar a execução de tarefas repetitivas.

#### 4.2 LEVANTAMENTO DE CONTEÚDO COM ENFOQUE NA ÁREA

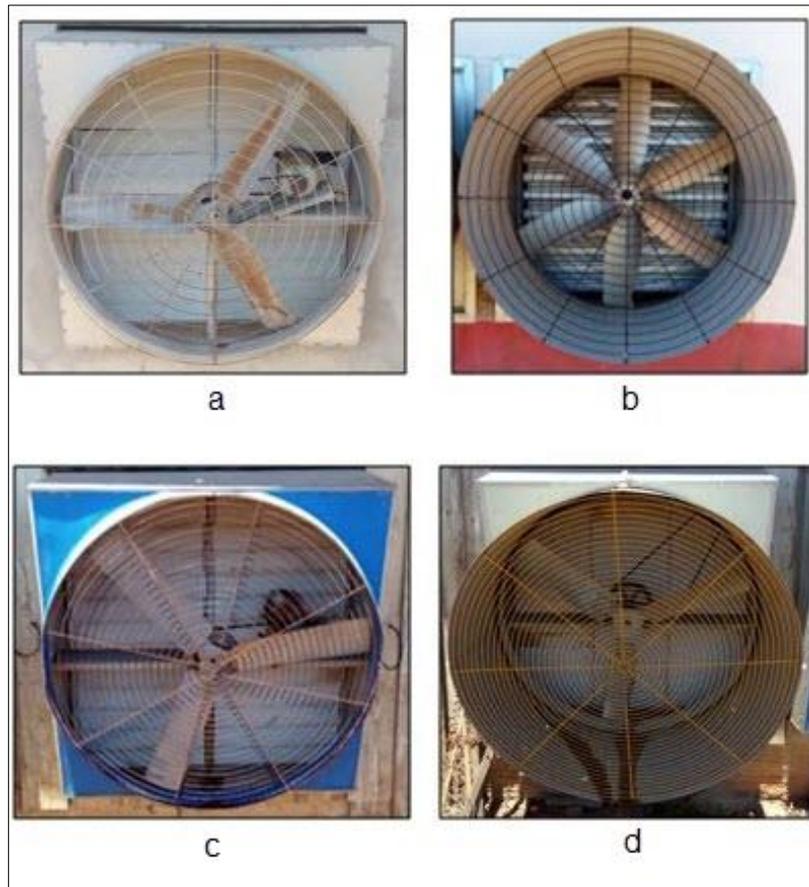
Realizou-se uma abrangente pesquisa em trabalhos realizados com o mesmo enfoque da área de estudo desse, buscando o máximo de informações possíveis que pode ser feito utilizando desta metodologia, além de buscar alternativas a serem utilizadas no desenvolvimento do projeto estipulado. Estas pesquisas dizem respeito à (aos):

- a) Importância da área estudada regional, nacional e mundialmente;
- b) A indústria e os equipamentos utilizados na avicultura, com enfoque na fabricação de equipamentos de exaustão;
- c) Utilização de *softwares* de CAD 3D nas áreas agrícolas e a relevância que estes projetos têm para a ciência como um todo;
- d) *Softwares* envolvidos no trabalho, buscando sua utilização específica no enfoque do projeto realizado;
- e) Aplicações realizadas com as ferramentas utilizadas neste trabalho na indústria como forma geral.

#### 4.3 DEFINIÇÃO DE MODELO PADRÃO E MODELAGEM DO EXAUSTOR

Primeiramente foram adquiridas imagens de diferentes modelos de exaustores utilizados em aviários para que fosse definido um modelo padrão a ser desenvolvido como base para a programação e automação desses procedimentos. A Figura 12 apresenta quatro modelos de exaustores dos quais foram coletadas imagens no intuito de definir qual seria utilizado como modelo padrão. Por fim selecionou-se o exaustor que utiliza um cone frontal para canalização do ar retirado do interior do aviário que,

de acordo com Carniato et al. (2015), possui capacidade nominal de movimentação de ar superior aos modelos de exaustores que não utilizam este componente, enquanto os modelos que utilizam o cone frontal possuem capacidade média de vazão de até  $36.650\text{m}^3/\text{h}$ , os modelos sem o cone frontal possuem capacidade média de vazão de  $33.750\text{m}^3/\text{h}$ .



**Figura 12 - Exaustores analisados para serem utilizados como modelo padrão.  
Fonte: Autoria própria.**

Legenda:

- a) Exaustor de 3 pás curvas com acionamento por correia;
- b) Exaustor de 6 pás curvas com acionamento por correia e cone frontal;
- c) Exaustor de 3 pás retas com acionamento por correia;
- d) Exaustor de 3 pás curvas com acionamento por correia e cone frontal;

Com o modelo já definido e as medidas e imagens dos exaustores já adquiridas foi possível realizar as modelagens dos componentes utilizando as dimensões

coletadas e as imagens dos componentes. As Figuras 13 e 14 mostram exemplos de imagens coletadas dos componentes, neste caso são da polia e da grade frontal, respectivamente.



**Figura 13 - Polia utilizada como modelo.  
Fonte: Autoria própria.**



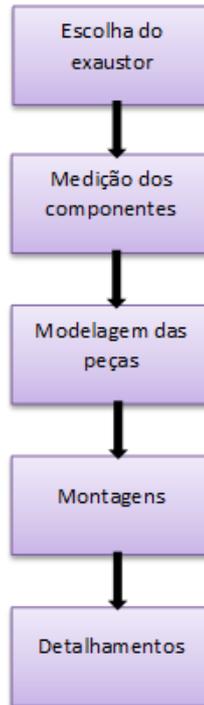
**Figura 14 - Grade frontal utilizada como modelo.  
Fonte: Autoria própria.**

Na Figura 15, pode-se ver o momento da verificação das medidas. Todas foram coletadas com valores aproximados, uma vez que o modelo foi utilizado apenas como base para modelagem de um equipamento que, posteriormente, teria suas medidas frequentemente modificadas de acordo com a necessidade do usuário.



**Figura 15 - Medição dos componentes.**  
Fonte: Autoria própria.

Para fazer a modelagem, foi utilizado o *software* SolidWorks®, sendo modeladas primeiramente todas as peças que compõem o exaustor, para então ser feita a montagem das peças formando o equipamento. O último passo da elaboração do projeto foi fazer o detalhamento das peças individuais e das montagens e todos estes passos podem ser verificados no fluxograma da Figura 16.



**Figura 16 - Fluxograma de execução do modelo inicial.**  
**Fonte: Autoria própria.**

Estes procedimentos são muito relevantes, por darem uma real perspectiva de como ficará o equipamento depois de fabricado e o detalhamento, em particular, é de grande importância para a indústria utilizar no momento da manufatura das peças, uma vez que ela mostra todas as medidas, agilizando a fabricação e evitando erros de execução.

#### 4.4 CRIAÇÃO E EDIÇÃO DOS CÓDIGOS PARA AUTOMAÇÃO

Com a modelagem pronta e feitas todas as alterações necessárias, o próximo procedimento realizado foi a criação dos códigos que executam as ações necessárias para criar as peças, montagens e detalhamento do exaustor. Primeiro foram escritos os códigos que irão modelar cada peça individualmente, para que, depois, seja feita a montagem destas peças e, por último, os detalhamentos.

Alguns dos procedimentos de modelagem no SolidWorks® foram refeitos, desta vez gravando as ações realizadas durante a modelagem em macros, que foram editadas com o intuito de auxiliar no processo de digitação dos códigos. Devido ao

fato de que quando as macros são gravadas geram muitas linhas desnecessárias, este procedimento foi utilizado, principalmente, para auxiliar na hora de definir as funções a serem utilizadas e agilizar o processo de digitação. Outra desvantagem de se utilizar exatamente o código gerado pelas macros é a declaração de variáveis, uma vez que estas são declaradas de modo genérico, o que faz o processo de depuração do código ser mais lento e o torna mais difícil de ser compreendido. Neste trabalho as variáveis foram declaradas o mais específico possível e o código foi organizado de modo a facilitar a leitura e o entendimento.

Todos os códigos depois de já escritos foram testados, de modo a detectar possíveis erros que pudessem ocorrer quando da utilização do usuário, simulando possíveis ações que este possa executar quando estiver utilizando o programa. A principal intenção destes testes foi fazer com que o código seja perfeitamente usual, alterando tipos e nomes de variáveis que sejam alteradas pelo usuário e definindo parâmetros e limites com os quais o programa deve funcionar.

Depois que todos os códigos que criam as peças individualmente estavam prontos e testados, estes foram unidos em um código principal, e cada um destes códigos foi criado de modo a trabalhar como módulo, que é uma forma de armazenar rotinas gerais e reutilizáveis dentro do programa (PERRY, 1999), sendo estes módulos “chamados” pelo procedimento principal, no momento em que devem ser executados.

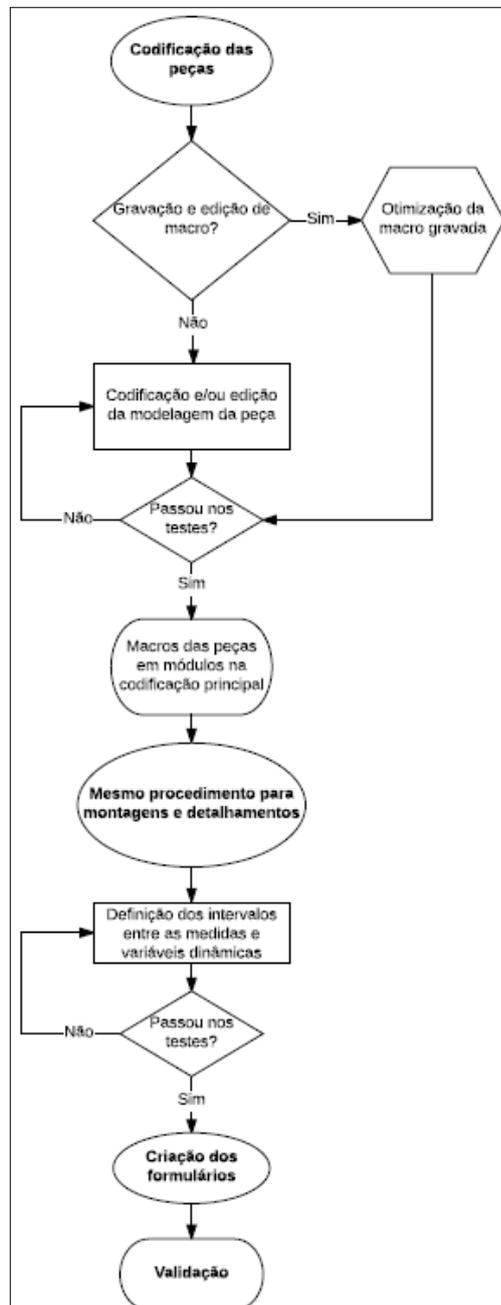
Com os códigos que realizam a modelagem de cada um dos componentes funcionando, foi feita a codificação das macros que geram as montagens e sub-montagens do equipamento, novamente escrevendo os códigos que fazem estes procedimentos, utilizando as funcionalidades de macro sempre que necessário e corrigindo erros que eventualmente ocorreram no momento da automação destes processos. Estes códigos de sub-montagens e montagens foram, depois de testados, adicionados ao código principal, como aconteceu com os códigos das peças individuais.

A seguir, foi feita a automação do processo de detalhamento das peças e montagens do exaustor. Para tal, foram utilizadas as mesmas técnicas citadas anteriormente, escrevendo e testando o código para, então, adicioná-lo ao código da macro principal e finalizar o processo de automação da modelagem.

Por fim, com o código principal pronto e funcionando corretamente, foi desenvolvida a interface, a qual é utilizada para interagir com o usuário, permitindo

que este altere os principais valores das medidas geométricas do exaustor conforme sua necessidade e o programa defina estes valores para suas respectivas variáveis e realize a modelagem partindo destas novas medidas definidas pelo utilizador. Dessa forma, um novo projeto será criado, desde a modelagem e montagem das peças, até o detalhamento a ser utilizado na manufatura, com as medidas solicitadas.

Todos os passos abordados nesta etapa estão demonstrados no fluxograma da Figura 17, onde é possível ter uma visão geral dos procedimentos adotados.



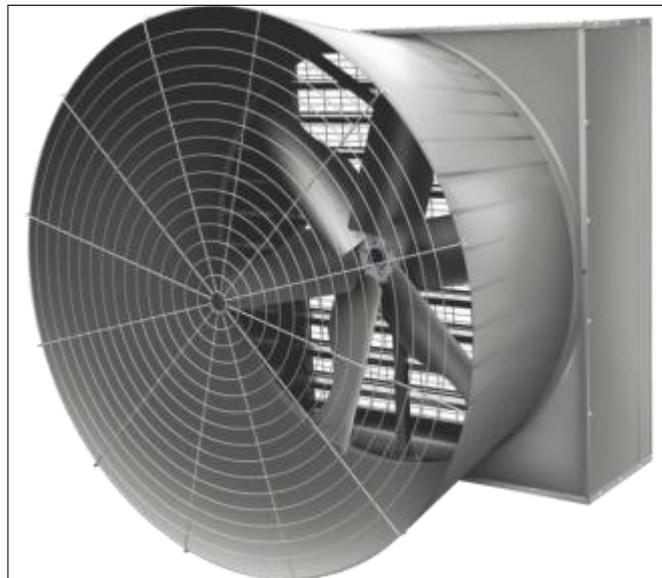
**Figura 17 - Fluxograma de execução das macros e formulários.**  
**Fonte: Autoria própria.**

## 5 RESULTADOS E DISCUSSÕES

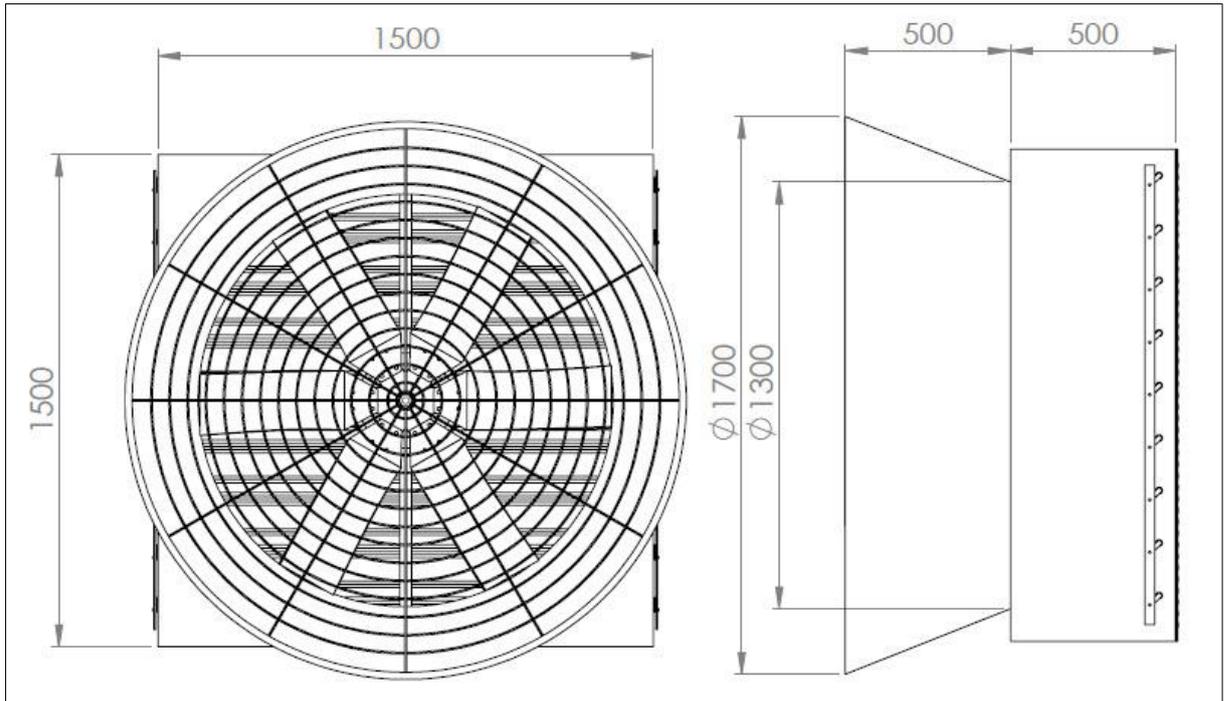
Nesta seção serão abordados os resultados obtidos com a realização da metodologia empregada, além de técnicas abordadas para conclusão da atividade.

### 5.1 COLETA DE INFORMAÇÕES E DEFINIÇÃO DE MODELO PADRÃO

Para definição do padrão de exaustor a ser modelado inicialmente, no intuito de determinar qual forma final o equipamento deverá ter, foram coletadas imagens de um modelo de exaustor utilizado na avicultura de corte e foram colhidas as principais medidas do equipamento selecionado. Estas medidas foram utilizadas para fazer a modelagem inicial do exaustor. Na Figura 18 é apresentado o modelo selecionado para ser utilizado como padrão e na Figura 19, vê-se as principais medidas utilizadas na confecção do modelo.



**Figura 18 - Modelo utilizado como padrão.  
Fonte: Avioeste (2016).**



**Figura 19 - Principais medidas do modelo.**  
**Fonte: Autoria própria.**

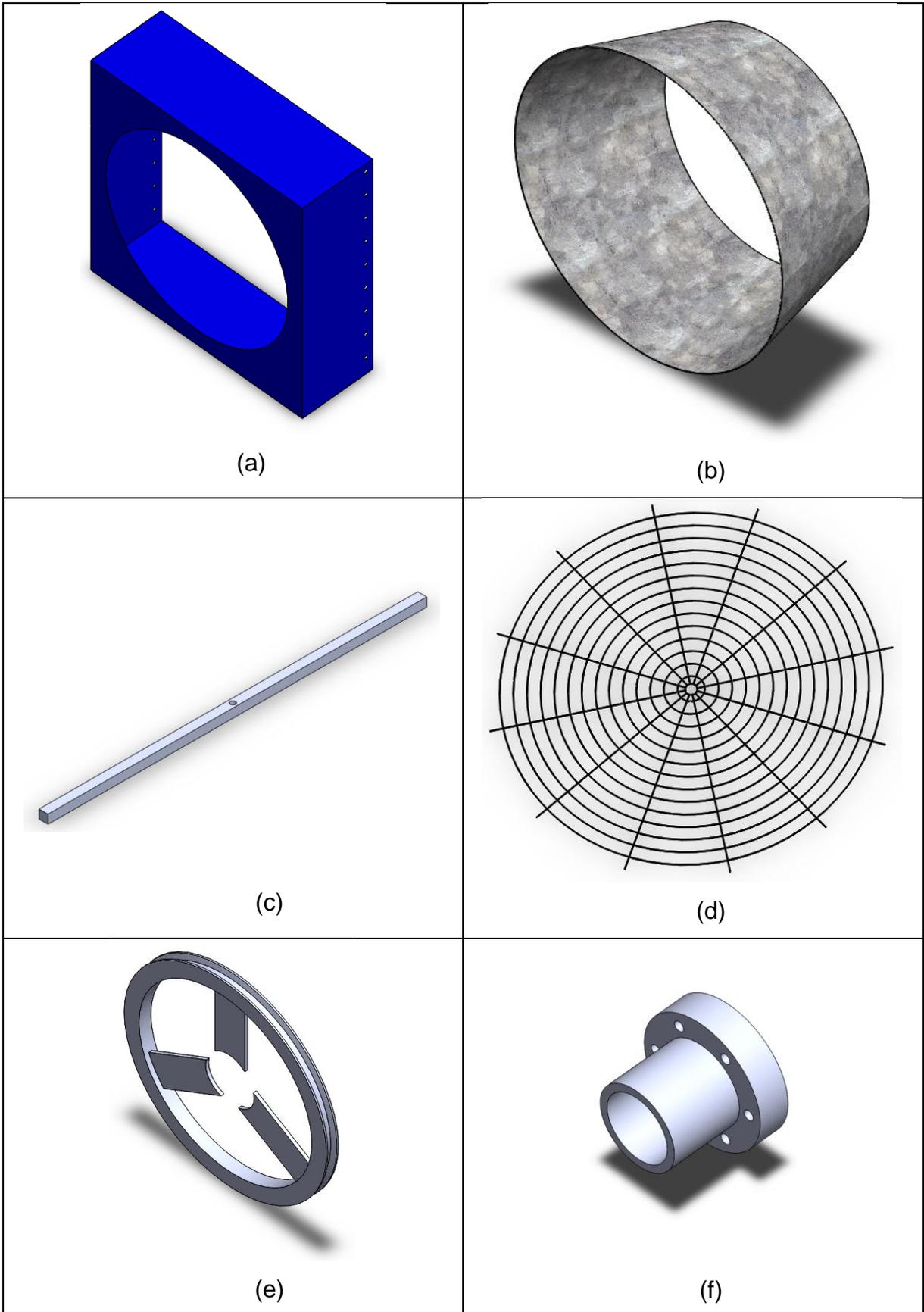
## 5.2 MODELAGEM DO EXAUSTOR MODELO

O SolidWorks® trabalha com três ambientes de desenvolvimento, o de peças, o de montagens e o de detalhamentos, e estas três etapas foram realizadas na modelagem do exaustor a ser utilizado como padrão no desenvolvimento das macros.

Tendo em mãos as medidas a serem utilizadas, foi possível iniciar a modelagem do exaustor. Utilizando o SolidWorks®, desenvolveu-se cada uma das peças que compõem o equipamento, representadas na sequência a seguir pelo Quadro 1.

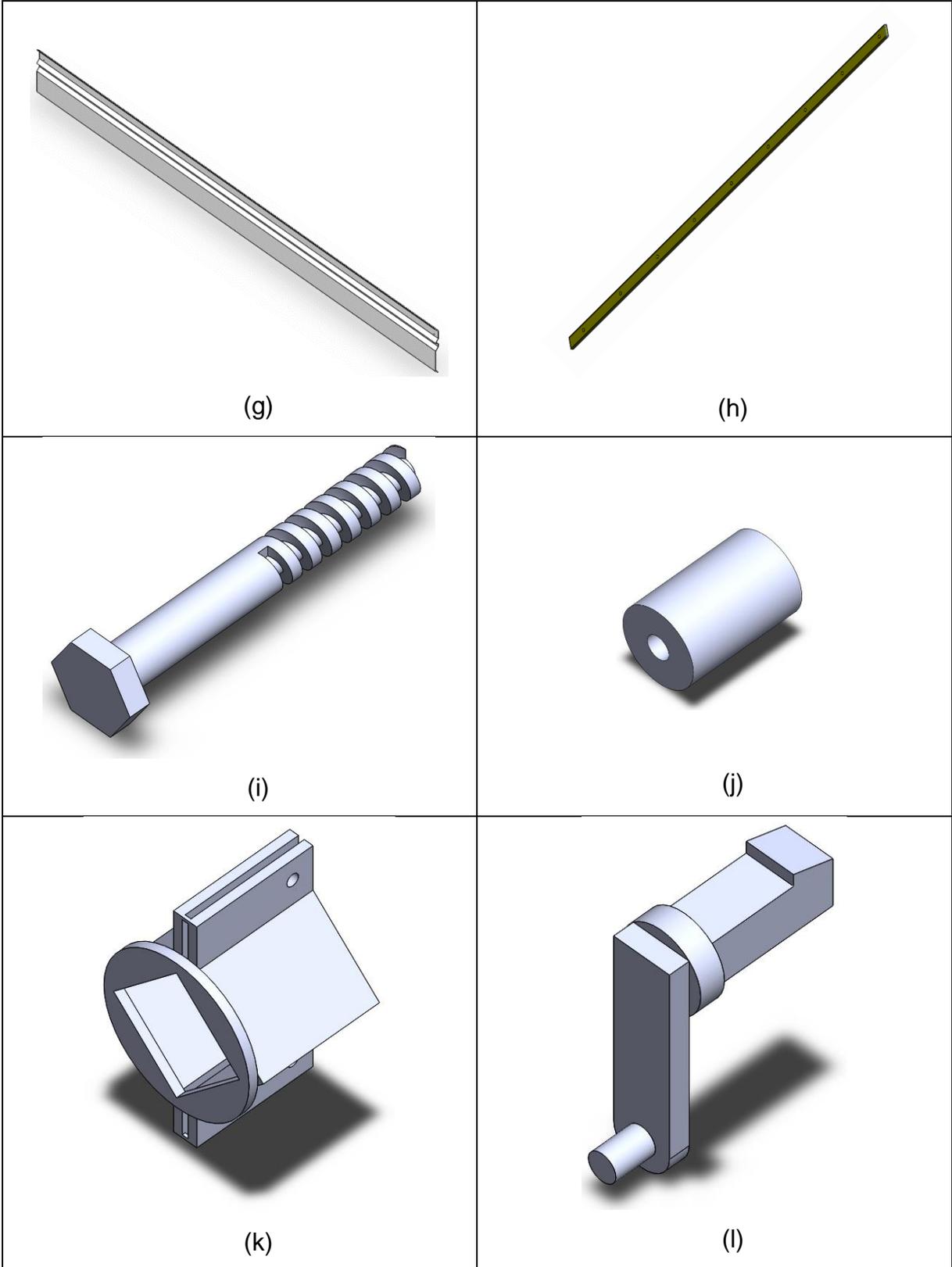
Quadro 1 – Peças desenvolvidas para exaustor modelo.

(Continua)



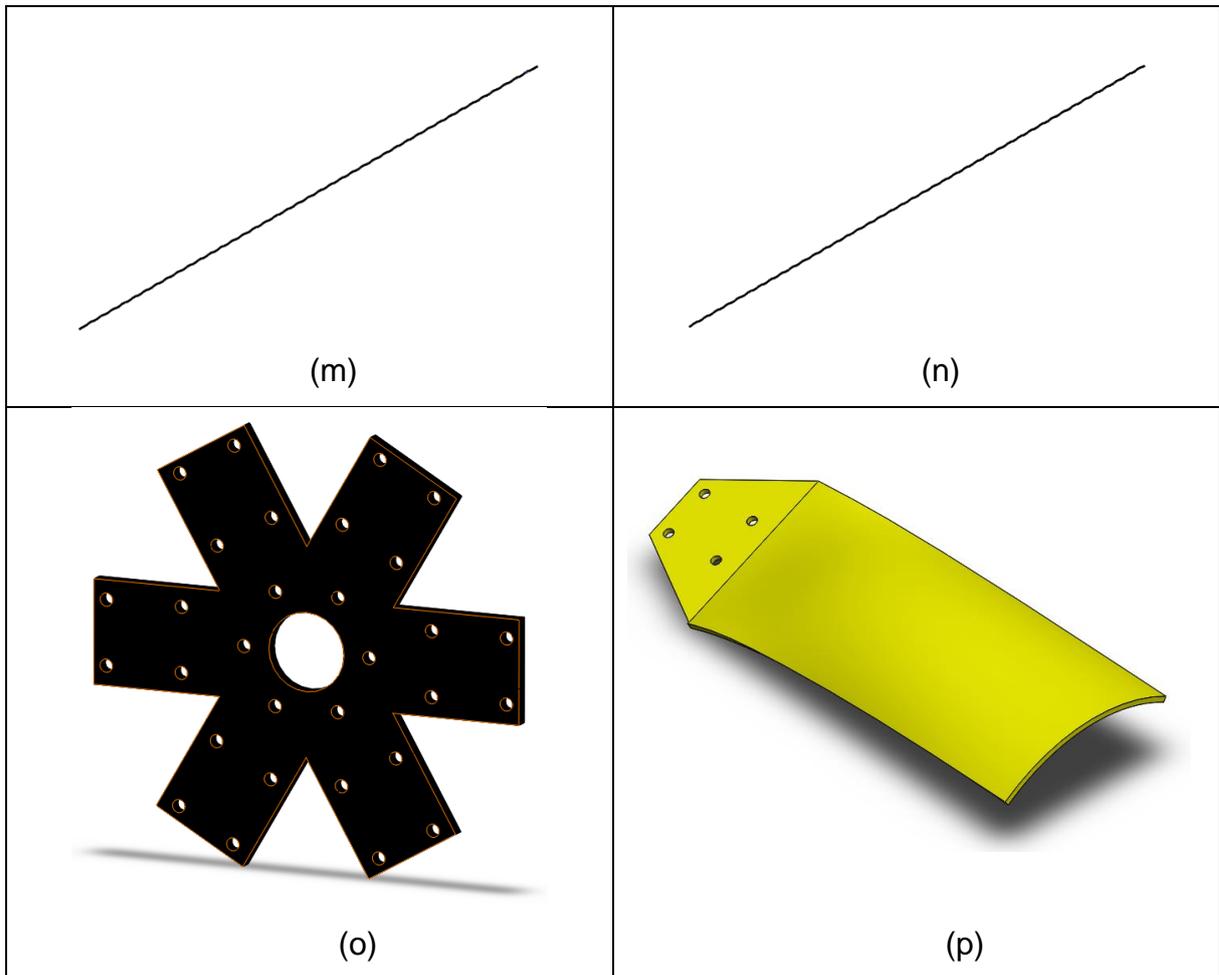
Quadro 1 – Peças desenvolvidas para exaustor modelo.

(Continua)



Quadro 2 - Peças desenvolvidas para exaustor modelo.

(Conclusão)



Fonte: Autoria própria.

Legenda:

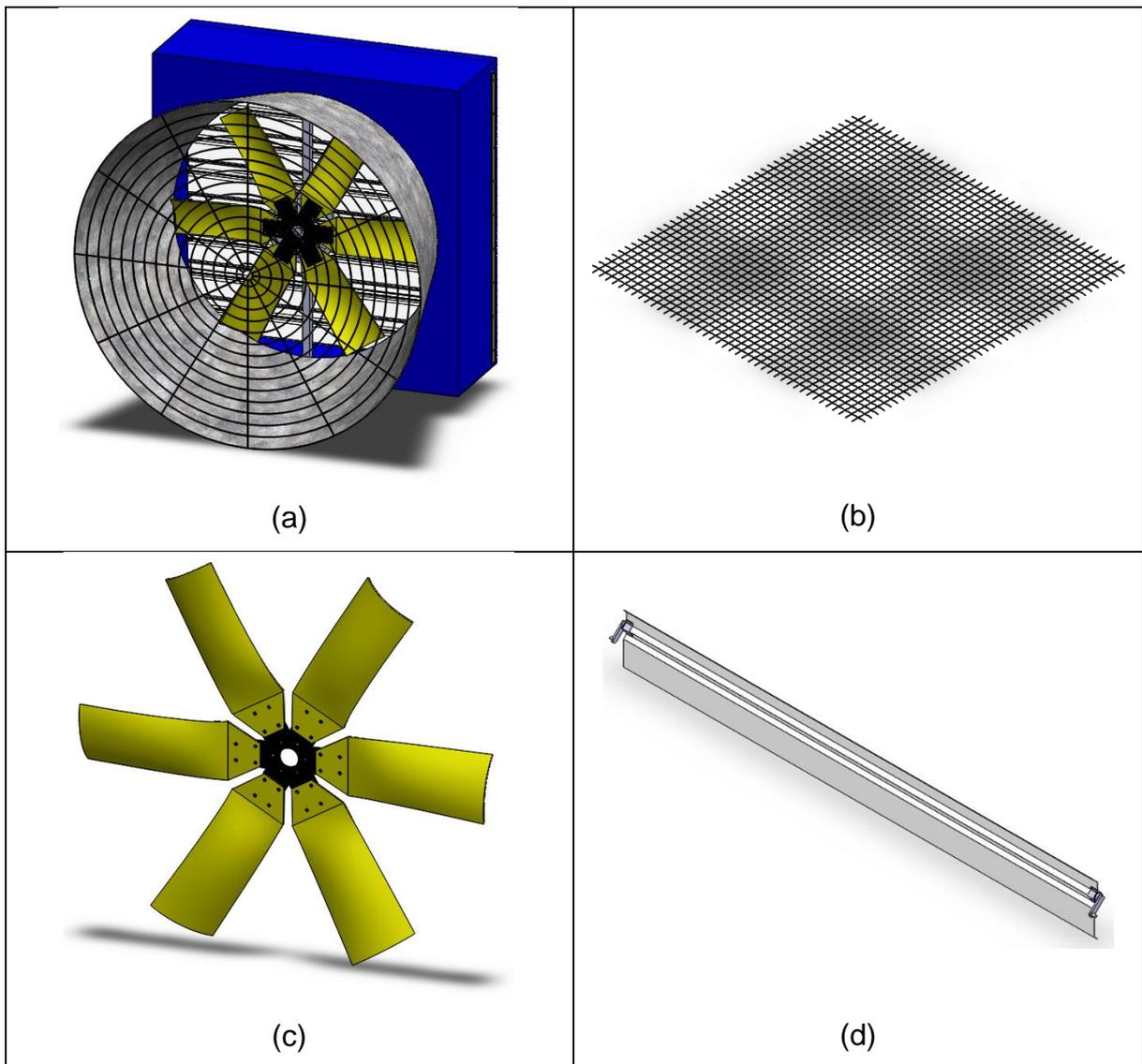
- a) Carcaça;
- b) Cone;
- c) Haste;
- d) Grade frontal;
- e) Polia;
- f) Fixador da polia;
- g) Veneziana;
- h) Suporte de travas das venezianas;
- i) Parafuso do rotor;
- j) Bucha da polia;
- k) Trava das venezianas;

- l) Trava para o conjunto das venezianas;
- m) Grade traseira 1;
- n) Grade traseira 2;
- o) Rotor;
- p) Hélice.

Após o desenvolvimento das peças realizaram-se, então, as sub-montagens e a montagem final do equipamento e estas são representadas no Quadro 2.

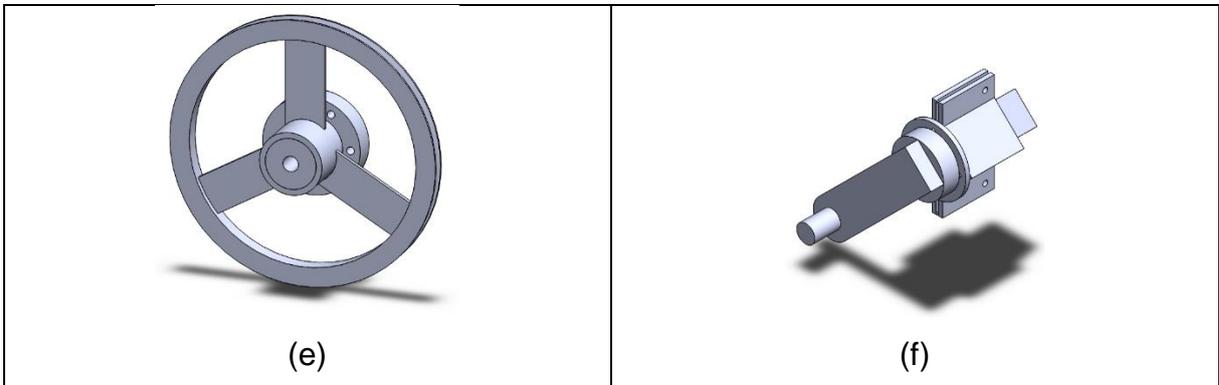
**Quadro 2 - Montagens desenvolvidas para exaustor modelo.**

(Continua)



**Quadro 2 - Montagens desenvolvidas para exaustor modelo.**

(Conclusão)



Fonte: Autoria própria.

Legenda:

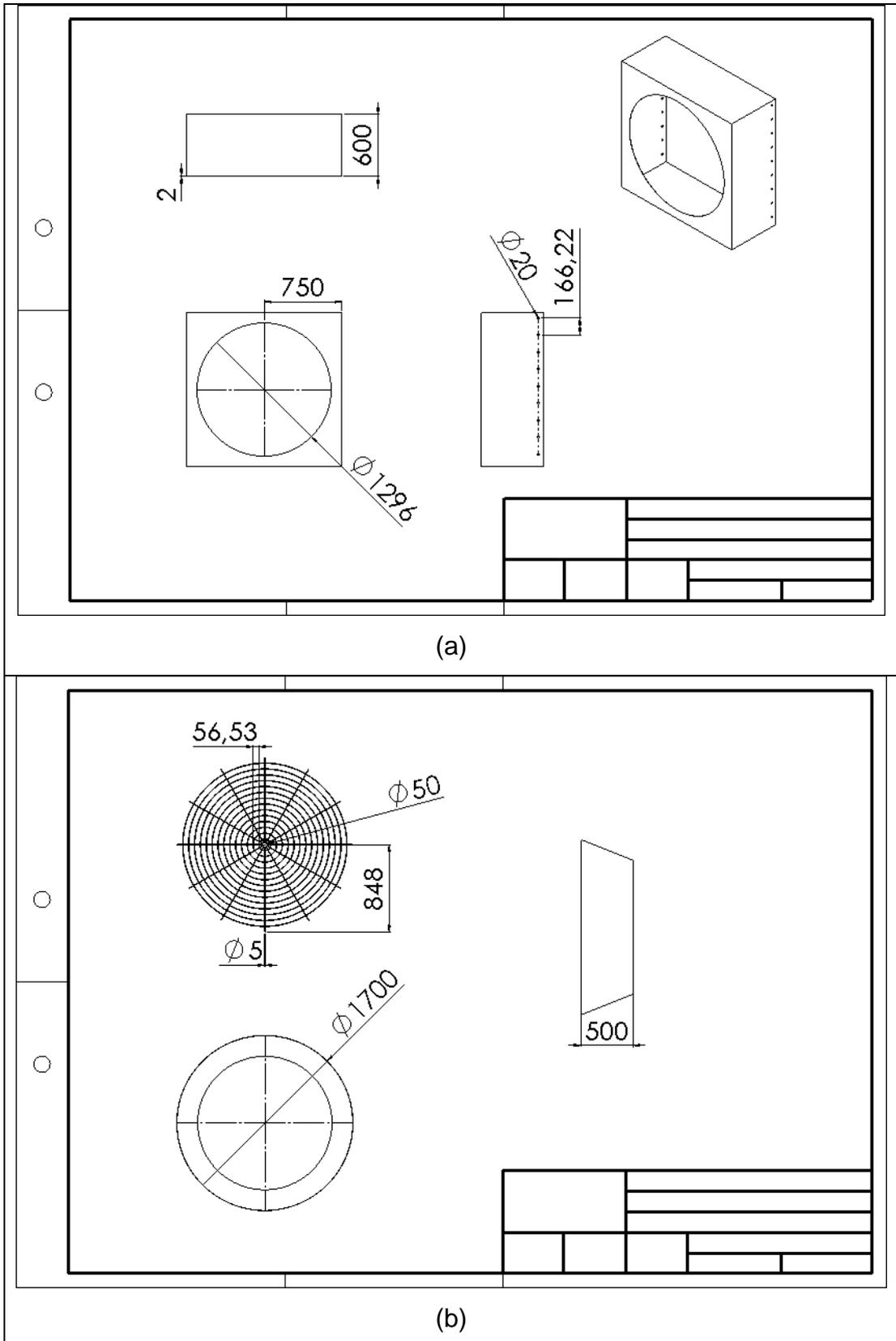
- a) Montagem final do exaustor;
- b) Montagem da grade traseira;
- c) Montagem do conjunto de hélices;
- d) Montagem das venezianas;
- e) Montagem da polia;
- f) Montagem das travas das venezianas.

Depois de realizados os ajustes necessários e definidos todos os recursos que seriam utilizados na elaboração do código, concluiu-se o desenvolvimento do equipamento a ser utilizado como modelo.

A fase seguinte de modelagem compreendeu no desenvolvimento das folhas de detalhamento do equipamento. Estas folhas serviram para saber como ficariam dispostas as peças e montagens em seus detalhamentos finais. Foi determinado como padrão de folha de desenho o tamanho A3 do padrão de folhas da ABNT (Associação Brasileira de Normas Técnicas), que possui 297 milímetros de altura por 210 milímetros de largura e foi suficiente para que todas as peças e montagens fossem detalhadas em um total de oito folhas de detalhamento. Cada uma destas oito folhas geradas inicialmente como modelo estão mostradas na sequência a seguir pelo Quadro 3.

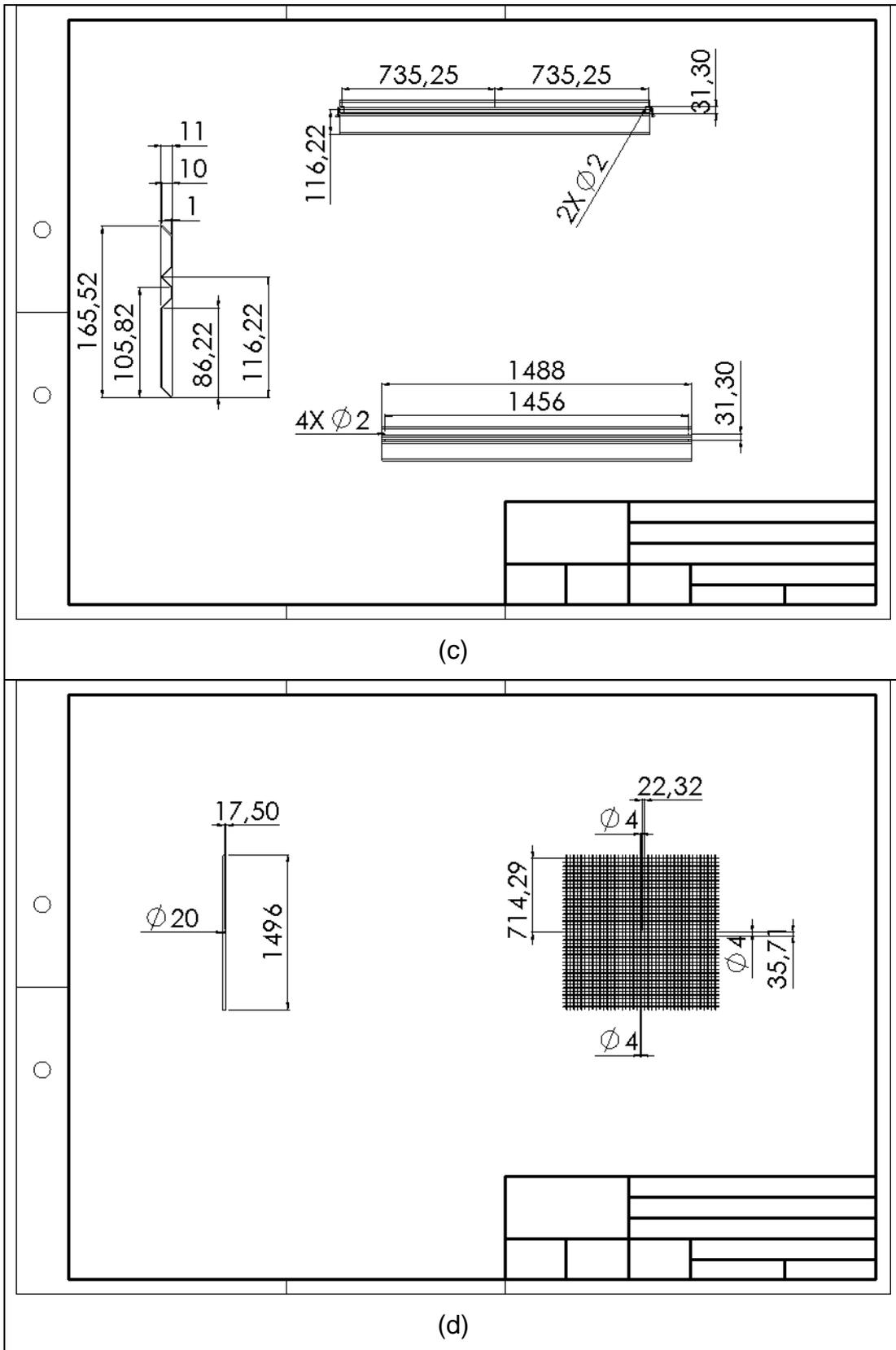
Quadro 3 – Esquema das folhas de detalhamento desenvolvidas para o exaustor modelo.

(Continua)



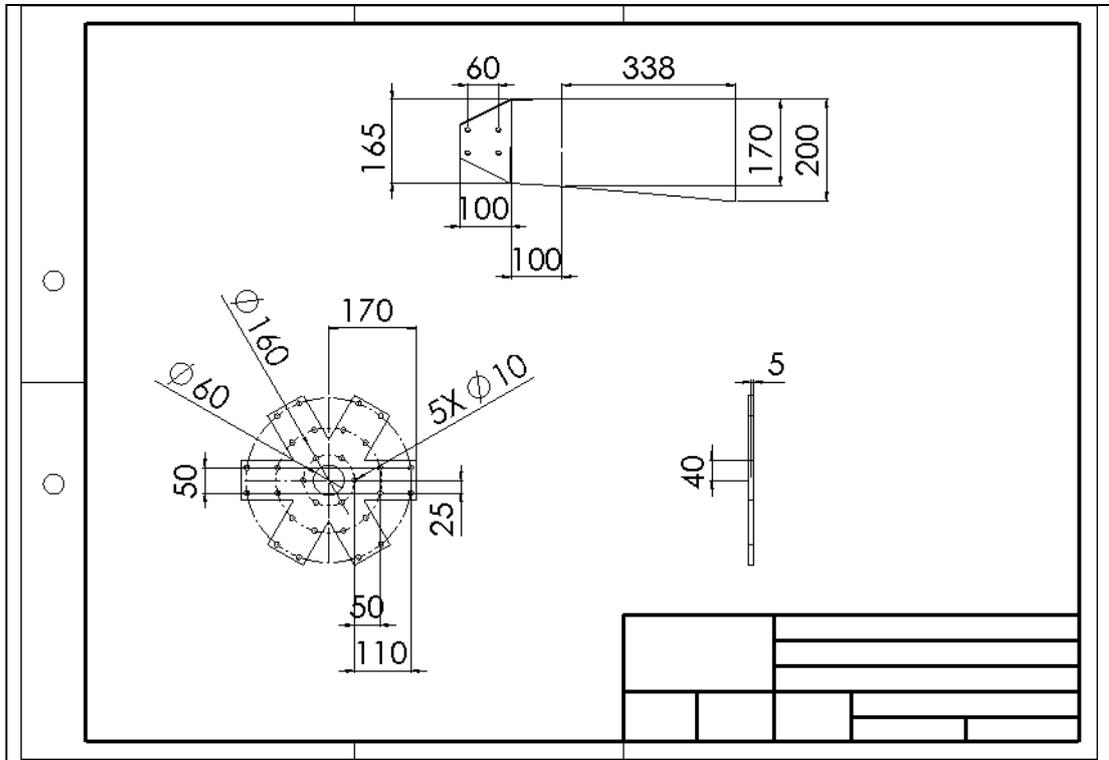
Quadro 3 - Esquema das folhas de detalhamento desenvolvidas para o exaustor modelo.

(Continua)

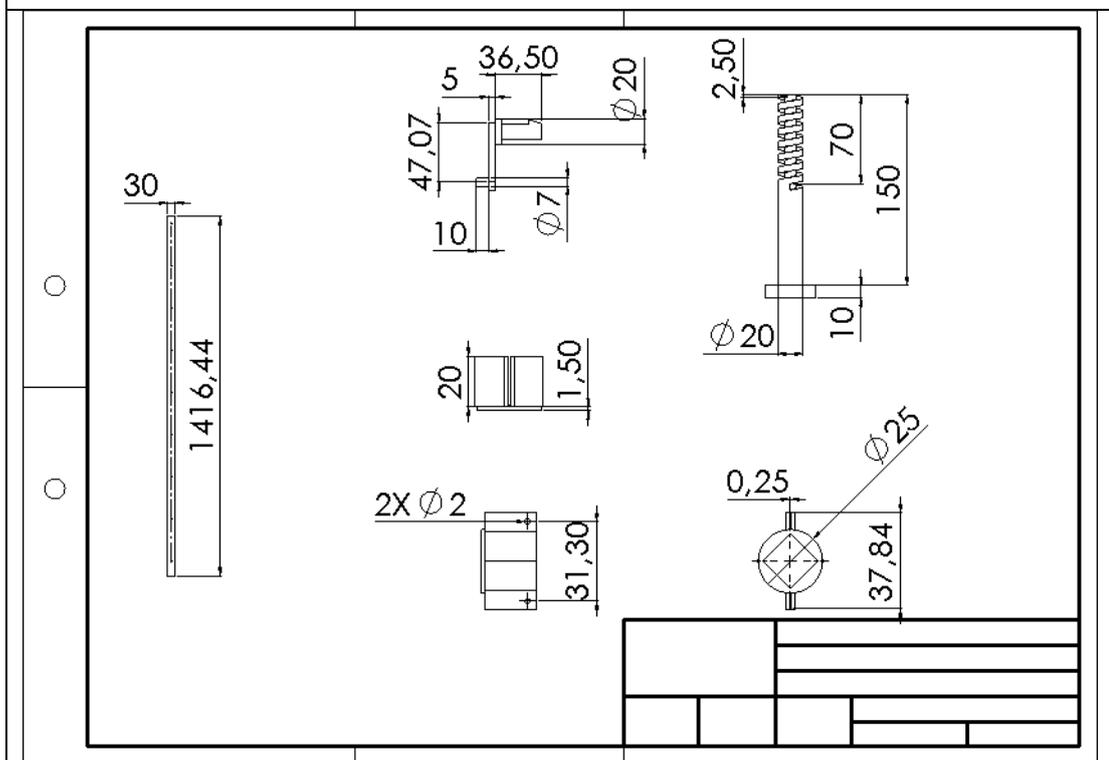


Quadro 3 - Esquema das folhas de detalhamento desenvolvidas para o exaustor modelo.

(Continua)



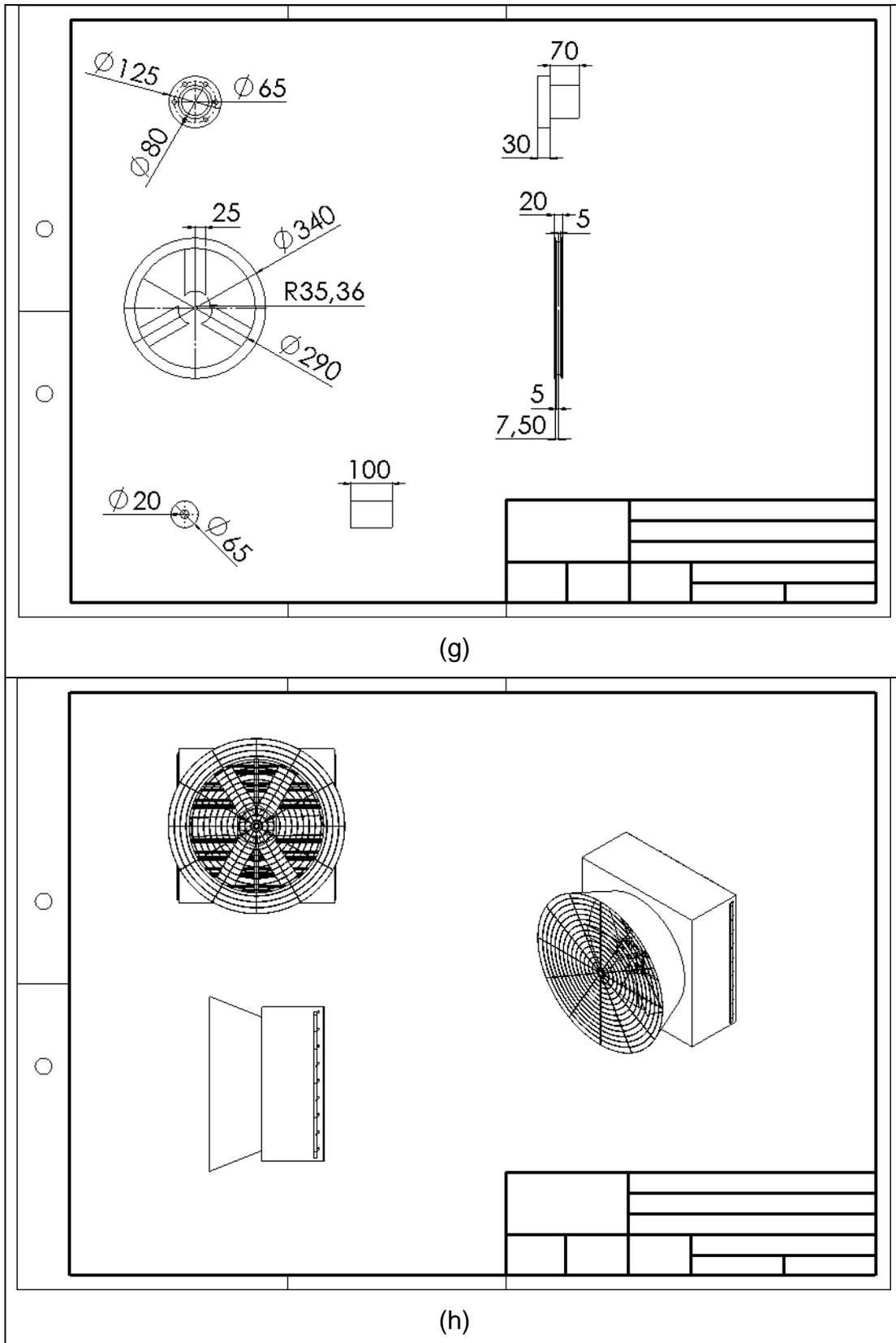
(e)



(f)

Quadro 3 - Esquema das folhas de detalhamento desenvolvidas para o exaustor modelo.

(Conclusão)



Fonte: Autoria própria.

Legenda:

- a) Carcaça;
- b) Cone e grade frontal;
- c) Venezianas;
- d) Haste de reforço e grade traseira;
- e) Rotor e hélices;
- f) Peças auxiliares;
- g) Polia;
- h) Montagem final do exaustor.

### 5.3 DESENVOLVIMENTO DA CODIFICAÇÃO

O desenvolvimento dos códigos responsáveis por executar a modelagem de todo o exaustor deu-se através da codificação de cada uma das macros individualmente e utilizando-se os recursos de gravação e edição da macro sempre que era necessário determinar um recurso ou uma medida específica a ser utilizada. Desta forma, agilizou-se o processo de codificação, uma vez que não foi necessário editar e otimizar o código gerado pela gravação da macro para todas as modelagens. Depois de feita a codificação da primeira peça, da primeira montagem e do primeiro detalhamento, seguiu-se o mesmo procedimento para a codificação dos demais componentes. Testava-se todos os códigos antes de se iniciar um novo componente.

#### 5.3.1 Desenvolvimento dos códigos responsáveis por gerar as peças

Primeiramente foram desenvolvidos os códigos responsáveis por gerar as peças que compõem o equipamento. No desenvolvimento da codificação do código que gera a carcaça, por exemplo, foi utilizado o recurso de gravação de macro e salvo todo o procedimento de abrir uma nova peça, iniciar um novo esboço e modelar o corpo principal da peça, não atentando-se às medidas que foram impostas, pois estas

seriam modificadas posteriormente já dentro do código de execução. Um trecho do código gerado por essa gravação pode ser visto na Figura 20.

```

Dim swApp As Object

Dim Part As Object
Dim boolstatus As Boolean
Dim longstatus As Long, longwarnings As Long

Sub main()

Set swApp = _
Application.SldWorks

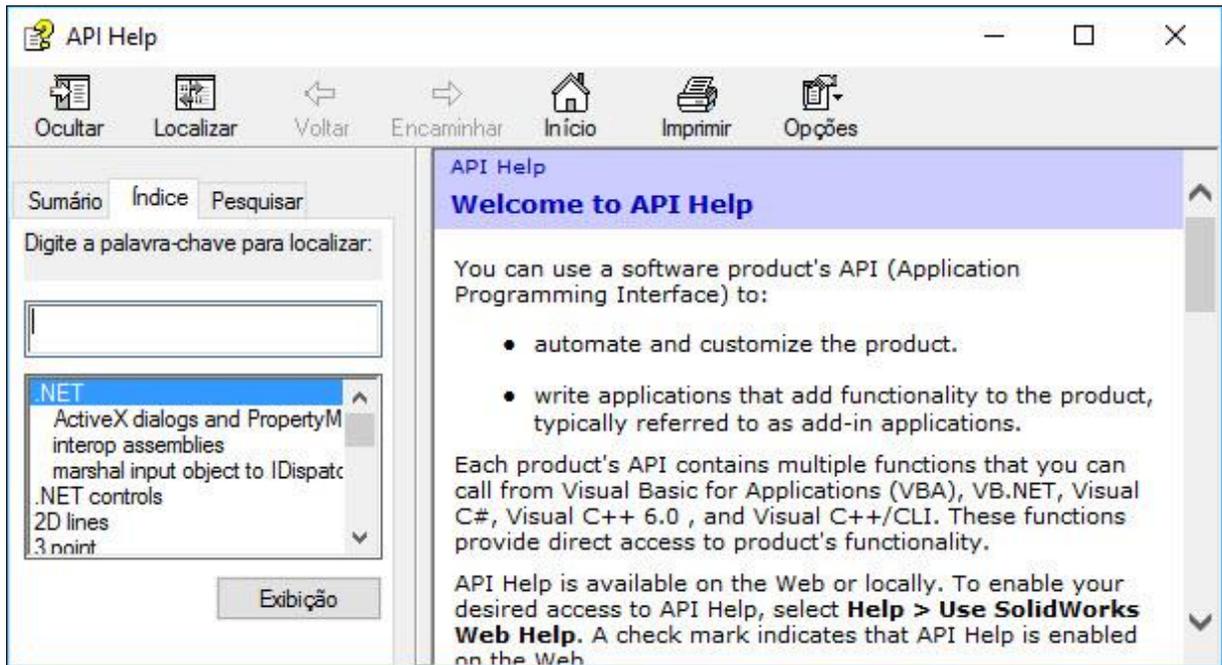
Set Part = swApp.NewDocument("C:\ProgramData\SolidWorks\SolidWorks
swApp.ActivateDoc2 "Peça1", False, longstatus
Set Part = swApp.ActiveDoc
Dim myModelView As Object
Set myModelView = Part.ActiveView
myModelView.FrameState = swWindowState_e.swWindowMaximized
boolstatus = Part.Extension.SelectByID2("Plano frontal", "PLANE",
Part.SketchManager.InsertSketch True
Part.ClearSelection2 True
Dim vSkLines As Variant
vSkLines = Part.SketchManager.CreateCenterRectangle(0, 0, 0, 4.294
Part.ClearSelection2 True
Part.SketchManager.InsertSketch True
End Sub

```

**Figura 20 - Trecho do código inicial de modelagem da carcaça.**  
**Fonte: Autoria própria.**

Muitos elementos do código gravado não são efetivamente importantes, como os trechos que fazem rotações de tela e trabalham com vistas no documento, por exemplo, e puderam ser eliminados sem prejuízo à execução, tornando o código mais simples e de mais ágil execução. Foram alterados, também, alguns métodos, seja para tornar o código menor, ou para torná-lo mais genérico, uma vez que este será executado em vários computadores diferentes e, portanto, é necessário evitar, por exemplo, a utilização de endereços específicos dentro do computador.

Os métodos e propriedades a serem alterados foram verificados diretamente na “API Help” do SolidWorks®, que é uma ferramenta disponibilizada pelo *software* que apresenta todos os recursos que pode-se manipular através da utilização das macros. A Figura 21 apresenta a interface inicial da “API Help”.



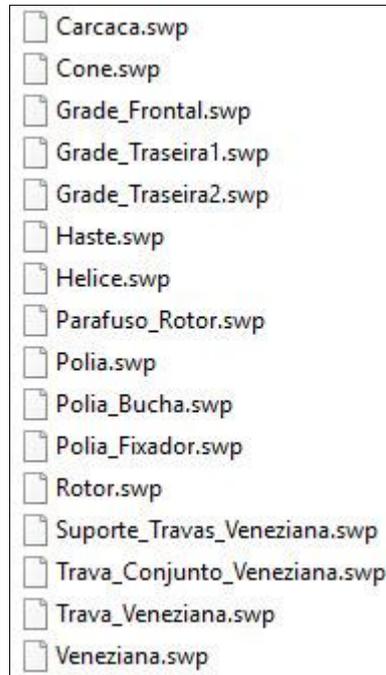
**Figura 21 - Interface inicial da API Help.**  
**Fonte: Autoria própria.**

Um método importante modificado é o método “NewDocument”, que foi alterado para o método “NewPart”, por este não exigir que se passe o endereço do modelo de documento a ser utilizado, trazendo automaticamente o modelo de documento padrão. Assim como este recurso, pesquisou-se e alterou-se todos os outros que forneciam problemas de compatibilidade e todas essas informações podem ser verificadas diretamente na API Help, já que é possível pesquisar por métodos que se quer utilizar e verificar quais os parâmetros que este determinado método exige que se passe a ele.

Com o desenvolvimento do código que gera a primeira peça, ficou mais fácil a codificação das peças posteriores, sendo possível codificar as novas macros diretamente na ferramenta de criação de macro, sem a necessidade de gravar e editar a mesma. O recurso de gravação e edição foi utilizado então, apenas para verificar como um método específico executa um determinado recurso.

Cada código de desenvolvimento das peças foi codificado e testado gradativamente, utilizando-se as medidas que foram definidas no exaustor modelo desenvolvido para esta finalidade.

Depois de escritos todos os códigos que geram cada uma das peças e estes serem testados, rodando e salvando as peças corretamente, gerou-se um arquivo de codificação de cada peça, conforme Figura 22.



**Figura 22 - Arquivos dos códigos que geram as peças.  
Fonte: Autoria própria.**

Estes códigos são apenas iniciais, eles foram codificados no intuito de definir e verificar o funcionamento dos métodos utilizados e, portanto, contém as medidas estáticas do exaustor. Os mesmos foram alterados posteriormente e ajustados para que utilizem variáveis ao invés de valores fixos e sejam alterados conforme a necessidade do usuário.

### 5.3.2 Desenvolvimento dos códigos responsáveis por gerar as montagens

Depois de concluídos os códigos que geram as peças, desenvolveu-se os códigos responsáveis por realizar as sub-montagens e montagem principal do exaustor.

O procedimento utilizado para isto foi parecido com o que foi feito para as peças, utilizando-se dos recursos de gravação e edição de macro para resolução de problemas pontuais e escrevendo o código na maioria das vezes, tornando-o mais simples e curto possível.

Na realização das montagens, a principal atividade realizada é o posicionamento das peças, que foram executadas utilizando métodos de seleção para, de fato, selecionar os componentes a serem posicionados e o método de realizar o posicionamento. Um detalhe importante a ser relatado é que a grande maioria dos posicionamentos foram feitos utilizando planos, sejam estes os que vêm por padrão no SolidWorks® ou os planos criados no decorrer das execuções. A Figura 23 apresenta um trecho de código que faz posicionamentos coincidente e de distância entre dois planos.

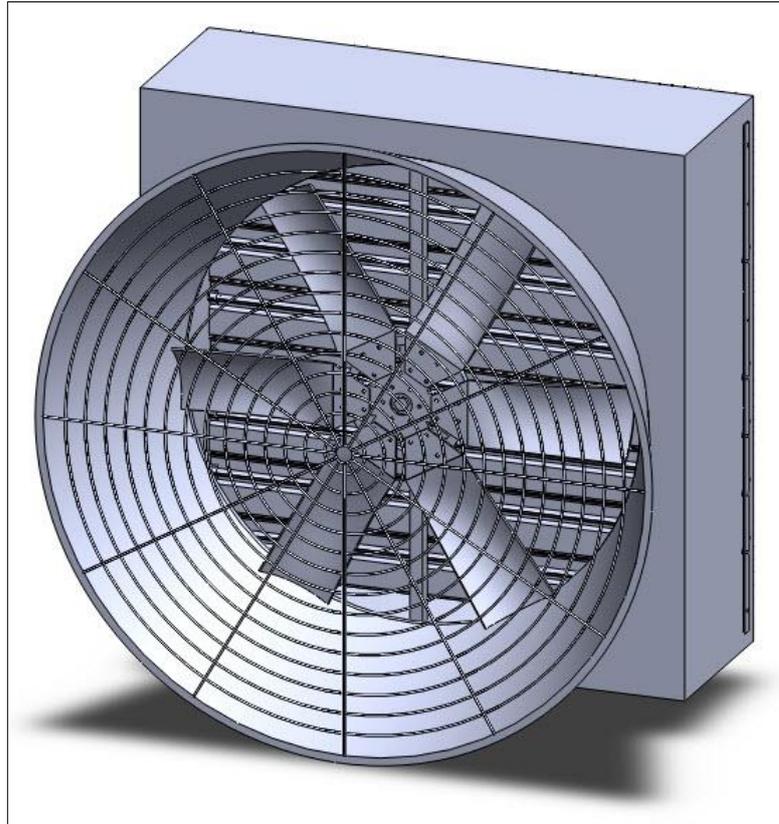
```
'Fazer o posicionamento de distância do cone em relação ao plano frontal da montagem
swModel.Extension.SelectByID2 strPlanoFrontal & "@" & swComp2.Name & "@" & swModel.GetTitle, _
"PLANE", 0, 0, 0, False, 0, Nothing, 0
swModel.Extension.SelectByID2 strPlanoFrontal, "PLANE", 0, 0, 0, True, 1, Nothing, 0
swAssy.AddMate3 5, 0, True, dblProfCarcaca / 2, dblProfCarcaca / 2, dblProfCarcaca / 2, _
0, 0, 0, 0, 0, False, Empty

'Posicionamento coincidente do plano superior e direito do cone com o plano superior e direito normal
swModel.Extension.SelectByID2 strPlanoSuperior & "@" & swComp2.Name & "@" & swModel.GetTitle, _
"PLANE", 0, 0, 0, False, 0, Nothing, 0
swModel.Extension.SelectByID2 strPlanoSuperior, "PLANE", 0, 0, 0, True, 1, Nothing, 0
swAssy.AddMate3 0, 1, False, 0, 0, 0, 0, 0, 0, 0, 0, 0, False, Empty
swModel.ClearSelection2 True
swModel.EditRebuild3

swModel.Extension.SelectByID2 strPlanoDireito & "@" & swComp2.Name & "@" & swModel.GetTitle, _
"PLANE", 0, 0, 0, False, 0, Nothing, 0
swModel.Extension.SelectByID2 strPlanoDireito, "PLANE", 0, 0, 0, True, 1, Nothing, 0
swAssy.AddMate3 0, 1, False, 0, 0, 0, 0, 0, 0, 0, 0, 0, False, Empty
swModel.ClearSelection2 True
swModel.EditRebuild3
```

**Figura 23 - Exemplo de codificação de posicionamento.**  
**Fonte: Autoria própria.**

Todas as sub-montagens e as peças remanescentes foram, então, utilizadas na montagem final do exaustor. Esta montagem pode ser verificada na Figura 24.



**Figura 24 - Montagem final do exaustor modelo.**  
**Fonte: Autoria própria.**

### 5.3.3 Desenvolvimento dos códigos responsáveis por gerar os detalhamentos

Por fim, gerou-se os códigos que fazem os detalhamentos das peças e das montagens desenvolvidas anteriormente. Estas folhas de detalhamento foram salvas, além do formato original do SolidWorks®, em formato PDF, na intenção de que estes possam ser abertos em computadores que não possuam, necessariamente, o SolidWorks®. Tem-se aqui novamente algumas peculiaridades em relação aos códigos das peças e das montagens como a utilização de interfaces e métodos de manipulação de folhas de desenho e anotações. Na Figura 25 tem-se demonstrado o exemplo de um trecho da codificação utilizada para gerar os detalhamentos e na Figura 26 é possível verificar alguns exemplos das folhas de detalhamento geradas.

```

'Criar um novo desenho
strTemplatePath = swApp.GetUserPreferenceStringValue _
(swUserPreferenceStringValue_e.swDefaultTemplateDrawing)
Set swModel = swApp.NewDocument(strTemplatePath, 8, Empty, Empty)
Set swDraw = swModel
Set swSketchMgr = swModel.SketchManager

'Alterar o nome das folhas
Dim swSheet As SldWorks.Sheet
Dim vSheetNames As Variant
vSheetNames = swDraw.GetSheetNames
Set swSheet = swDraw.Sheet(vSheetNames(0))
swSheet.SetName "Exaustor"

If strFormatPath <> "" Then
    swDraw.ActivateSheet "Exaustor"
    swDraw.SetupSheet5 swSheet.GetName, 8, 12, 1, 1, False, _
    strFormatPath, Empty, Empty, "Default", True
End If

'Adicionar 3 vistas de desenho e vista isométrica
Dim swView As SldWorks.View
swApp.DocumentVisible False, 1
swApp.OpenDoc6 strModelPath, 1, 0, "", Empty, Empty
swApp.DocumentVisible True, 1
swDraw.CreateDrawViewFromModelView3 strModelPath, "*Frontal", 0, 0, 0
swDraw.CreateDrawViewFromModelView3 strModelPath, "*Direita", 0, 0, 0
swDraw.CreateDrawViewFromModelView3 strModelPath, "*Isométrica", 0, 0, 0
swSheet.SetScale 1, 20, False, False

'Posicionar vistas
Dim vViews As Variant
vViews = swSheet.GetViews
Set swView = vViews(0)
Dim vPosition(1) As Double
vPosition(0) = 0.115
vPosition(1) = 0.225
swView.Position = vPosition
Dim dblescala(1) As Double
If dblCompCarcaca > 1.1 Then
    dblescala(0) = 1
    dblescala(1) = 30
Else
    dblescala(0) = 1
    dblescala(1) = 20

```

Figura 25 - Exemplo de codificação dos detalhes.  
Fonte: Autoria própria.

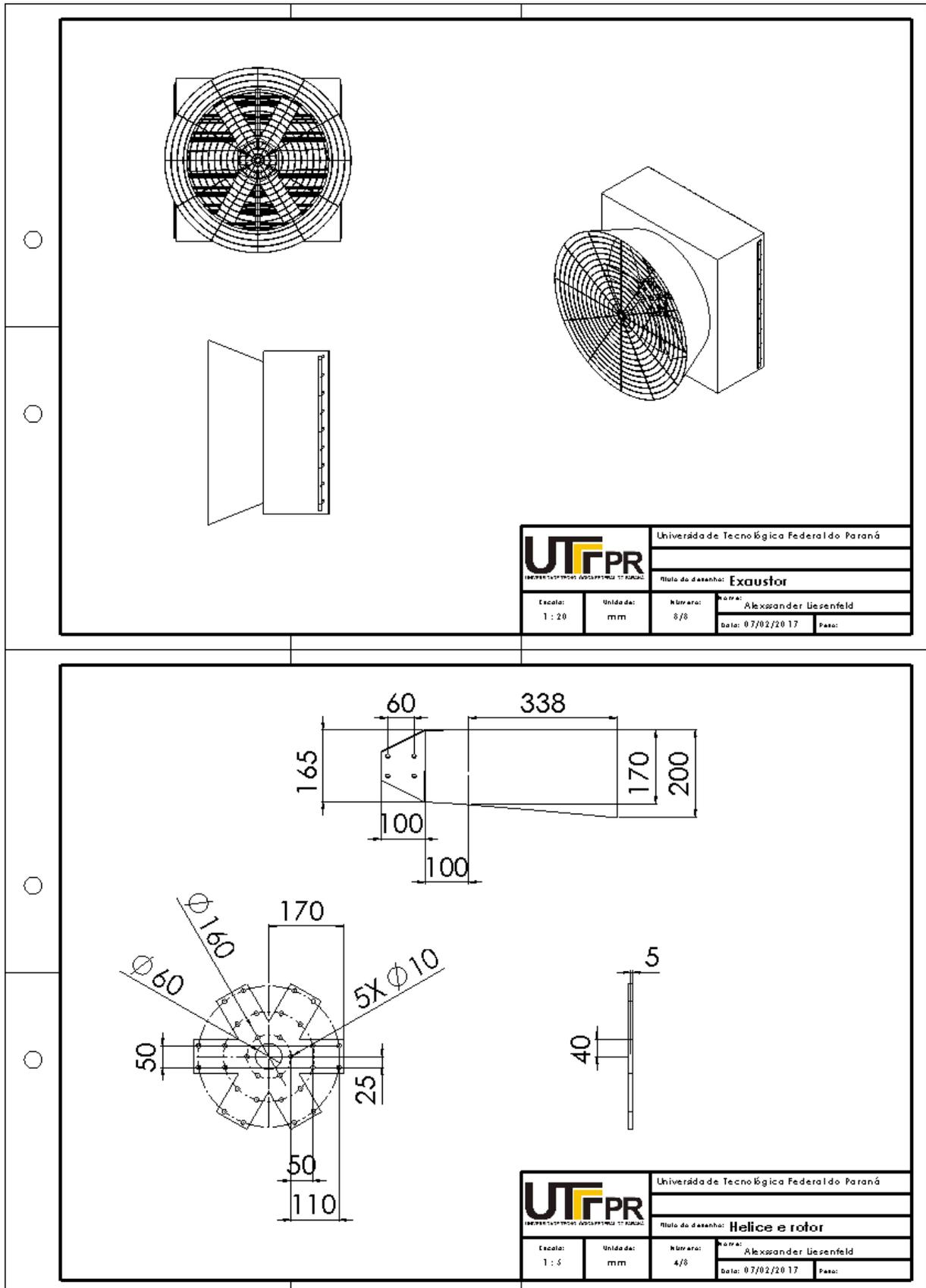


Figura 26 – Exemplos de detalhamentos gerados pela execução da macro.  
Fonte: Autoria própria.

Com o desenvolvimento dos códigos que geram os detalhamentos concluído, finalizou-se toda a etapa de codificação a ser utilizada como base para a automação da modelagem do exaustor.

#### 5.3.4 Utilização de variáveis na automação dos códigos

O exaustor modelado utilizando os códigos desenvolvidos possui medidas fixas, o que torna o projeto pouco usual. Devido a isto, os códigos foram modificados de modo a utilizar, ao invés de valores fixos, variáveis que podem ser modificadas conforme necessário. Algumas dessas variáveis serão definidas diretamente pelo usuário, outras servirão como auxiliares e serão alteradas automaticamente de acordo com as medidas definidas pelo utilizador. As variáveis que foram criadas para serem definidas pelo usuário são exibidas na Figura 27.

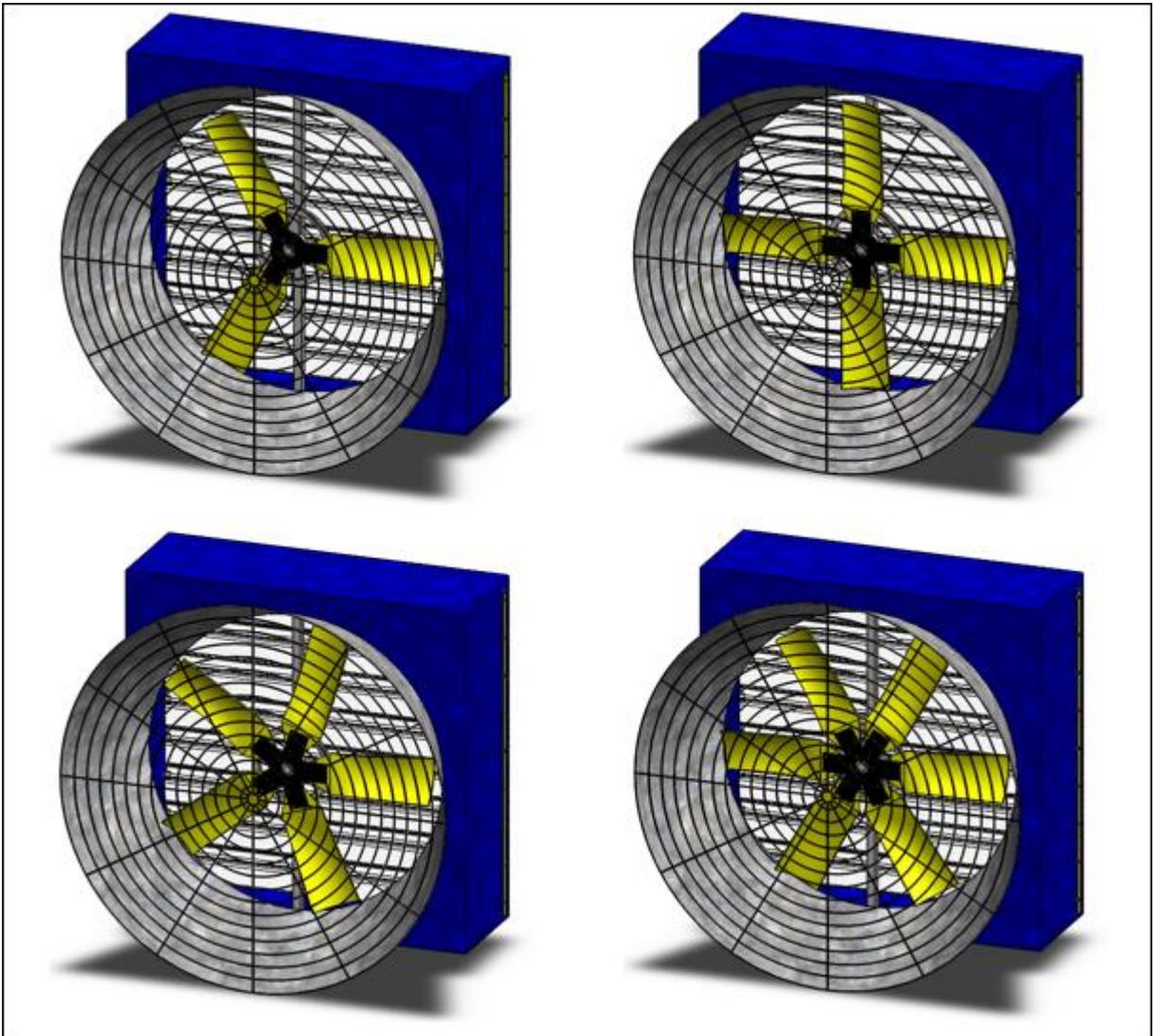
'Variáveis a serem definidas pelo usuário	
Public strFormatPath As String	'Definir o diretório do formato da folha
Public strDiretorio As String	'Definir o diretório para o qual salvar os arquivos
Public dblCompCarcaca As Double	'Definir o comprimento da carcaça
Public dblProfCarcaca As Double	'Definir a profundidade da carcaça
Public dblEspessura As Double	'Definir a espessura do material
Public dblCompCone As Double	'Definir o comprimento do cone
Public intHelices As Integer	'Definir a quantidade de hélices

**Figura 27 - Declaração das variáveis a serem definidas pelo usuário.**

As medidas a serem definidas são referentes à largura e profundidade da carcaça, profundidade do cone, espessura do material e número de hélices. Na Figura 28, pode-se ver quatro modelos desenvolvidos pela macro com as mesmas medidas geométricas mas com números de hélices diferente: três, quatro, cinco e seis, respectivamente. O tempo de execução do código varia de acordo com as configurações técnicas da máquina na qual o *software* está sendo executado, porém, normalmente leva cerca de cinco minutos para concluir toda a execução.

Além destes dados numéricos, devem ser definidas informações de nome do projetista e da empresa, diretório para salvamento do projeto e formato de folha a ser

utilizado, sendo que, além das medidas, apenas o dado referente ao diretório de salvamento é obrigatório.



**Figura 28 - Modelos com número de hélices diferente.**  
**Fonte: Autoria própria.**

Com os dados definidos e armazenados nas suas determinadas variáveis, estes foram sendo testados alterando as medidas dos mesmos e verificando todos os pontos nos quais estes influenciam. Assim, foram realizados os ajustes necessários para que estas medidas influenciadas pelas primeiras, também se alterem conforme a necessidade. Um exemplo disso está na extrusão da haste utilizada como reforço para o interior da carcaça do exaustor. Devido à largura da carcaça e a espessura do material serem dinâmicos, o comprimento desta haste deve se ajustar de acordo com essas medidas que o usuário deve definir, assim, o cálculo feito para que o

comprimento desta haste seja correto é o valor definido para o comprimento da carcaça, subtraído da espessura do material presente nos dois lados da carcaça. Na Figura 29 está destacado o cálculo citado já na definição do valor a ser utilizado na extrusão da peça.

```
swFeatureMgr.FeatureExtrusion2 True, False, False, 6, 0, ((dblCompCarcaca * 2) - dblEspessura - dblEspessura), 0, False, False, False, False, 0, 0, False, False, False, False, True, True, True, True, 0, 0, False
```

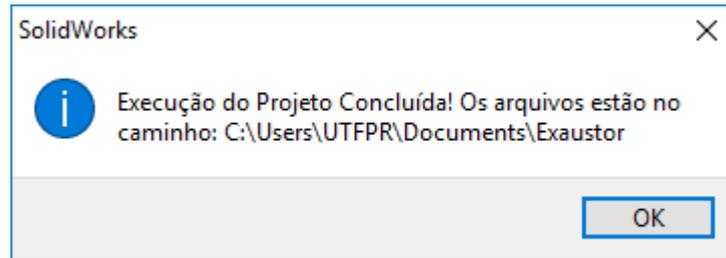
**Figura 29 - Definição de medida dinamicamente.**  
**Fonte: Aatoria própria.**

É importante salientar que cada uma das variáveis de definição de medidas é ajustada logo que o código principal é iniciado, já que as medidas que o usuário define são em milímetros e o SolidWorks® exige que estas sejam em metros. Na Figura 30 nota-se esta alteração, onde o valor informado pelo usuário é convertido de milímetros para metros dividindo-os por '1000' e este valor ajustado é definido para a variável correspondente.

```
dblCompCarcaca = (txtAltura.Value / 2) / 1000
dblCompCone = txtCone.Value / 1000
intHelices = txtHelices.Value
dblEspessura = txtMaterial.Value / 1000
dblProfCarcaca = txtProfundidade.Value / 1000
```

**Figura 30 - Mudança dos valores de milímetros para metros.**  
**Fonte: Aatoria própria.**

Os dados a serem manipulados pelo usuário foram submetidos a bloqueios de modo que possam ser definidos dentro de um determinado intervalo. Caso o usuário insira valores inválidos ou não os insira, é enviada uma mensagem instruindo-o de que estes devem ser modificados ou adicionados. Também é apresentado um aviso ao utilizador quando o projeto é concluído e qual o local onde as peças foram salvas, o qual está sendo mostrado na Figura 31. O código que realiza estes bloqueios e apresenta os avisos pode ser visto na Figura 32.



**Figura 31 - Aviso de execução do projeto concluída.**  
**Fonte: Autoria própria.**

```
'Fazer os bloqueios de valores
If (dblCompCarcaca < 0.6) Or (dblCompCarcaca > 1.5) Or (dblProfCarcaca < 0.2) Or _
(dblProfCarcaca > (txtAltura.Value / 1000)) Then
    swApp.SendMsgToUser2 "Valores para a carcaça estão fora dos limites estabelecidos.", 2, 2
    Call Codigo_Inicial
End
End If

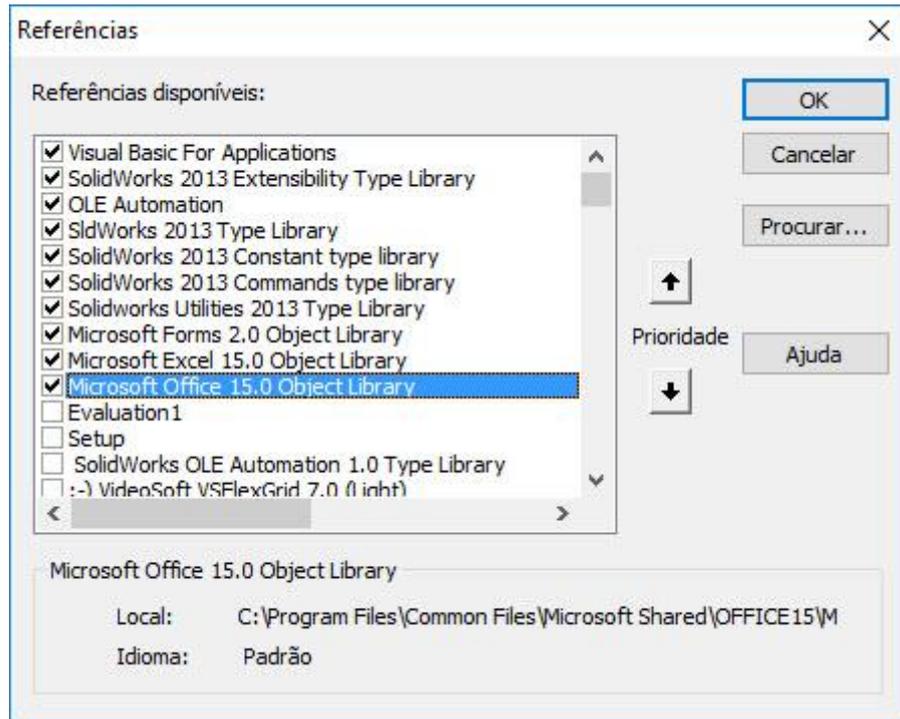
If (dblCompCone < 0.12) Or (dblCompCone > (dblProfCarcaca * 2)) Then
    swApp.SendMsgToUser2 "Valores para o cone estão fora dos limites estabelecidos.", 2, 2
    Call Codigo_Inicial
End
End If

If (dblEspessura < 0.001) Or (dblEspessura > 0.003) Then
    swApp.SendMsgToUser2 "Valores para a espessura da chapa estão fora dos limites estabelecidos.", 2, 2
    Call Codigo_Inicial
End
End If

If (intHelices < 2) Or (intHelices > 8) Then
    swApp.SendMsgToUser2 "Numero de hélices fora dos limites estabelecidos.", 2, 2
    Call Codigo_Inicial
End
End If
```

**Figura 32 - Trecho de código responsável por delimitar as medidas.**  
**Fonte: Autoria própria.**

Para seleção do local de salvamento dos arquivos e do formato de folha, foi utilizado o método "FileDialog", que é um recurso disponível no VBA através de uma biblioteca do Microsoft Office® 2013 que foi ativada conforme exibido na Figura 33.



**Figura 33 - Adição de referência ao Microsoft Office® 2013.**  
**Fonte: Autoria própria.**

É importante ressaltar que o recurso de selecionar um arquivo, como no caso de seleção do formato de folha a ser utilizado, ou o recurso de selecionar um caminho específico utilizando o explorador de arquivos do Microsoft Windows® é disponibilizado apenas utilizando esta biblioteca. Como já foi citado, neste trabalho a biblioteca utilizada pertence ao pacote Microsoft Office 2013®, portanto, caso o usuário possua uma versão diferente desta, deverá adicionar a biblioteca correspondente a sua versão à lista de referências do VBA antes de executar o código para que este funcione corretamente. O trecho de código responsável pela seleção do arquivo correspondente ao formato de folha e pela seleção do local de salvamento dos arquivos é exibido na Figura 34.

```

Public Sub SelecDiretorio_Click()

With windows.Application.FileDialog(msoFileDialogFolderPicker)
    .AllowMultiSelect = False
    .Title = "Selecionar diretório para salvar arquivos"
    .ButtonName = "Selecionar"
    .InitialFileName = "C:\\"
    If .Show = -1 Then 'Clicar em OK
        strDiretorio = .SelectedItems.Item(1)
        txtDiretorio.Value = strDiretorio
    End If
End With

End Sub

Public Sub SelecFolha_Click()
With windows.Application.FileDialog(msoFileDialogFilePicker)
    .AllowMultiSelect = False
    .Filters.Clear
    .Filters.Add "Formatos de Folha", "*.slddrt"
    .Title = "Selecionar diretório para salvar arquivos"
    .ButtonName = "Selecionar"
    .InitialFileName = "C:\\"
    If .Show = -1 Then 'clicar em OK
        strFormatPath = .SelectedItems.Item(1)
        txtFolha.Value = strFormatPath
    End If
End With

```

**Figura 34 - Código de seleção de formato de folha e diretório.**  
**Fonte: Autoria própria.**

Foram diversos os recursos utilizados em toda a codificação, sejam eles do VBA ou do próprio SolidWorks®. Todos os códigos de execução das peças individualmente, de cada montagem e sub-montagem e dos detalhamentos foi feito em módulos separados dentro do VBA, cada qual contendo as devidas particularidades. Todos estes códigos foram adicionados a uma única macro, juntamente com os formulários e o código inicial, que é responsável pela declaração das variáveis e por iniciar a execução do código a partir do formulário de abertura que, por sua vez, chama o formulário de inserção de dados e este executa o código que gera a primeira peça e, assim, sucessivamente até a conclusão do projeto.

Os módulos foram nomeados de acordo com a execução que geram dentro do SolidWorks®, diferenciando-se entre eles pela primeira letra, que indica em qual ambiente de modelagem este determinado módulo irá trabalhar. Módulos iniciados com a letra 'd' indicam que trata-se de um detalhamento, com a letra 'm' é uma

montagem e com a letra 'p' refere-se à uma peça. O código inicial foi representado pela letra 'a', apenas para que ficasse na primeira posição da lista de módulos. A Figura 35 mostra a árvore de projetos do VBA contendo todos os módulos e os dois formulários.



**Figura 35 - Árvore de projetos do VBA, depois da codificação finalizada.  
Fonte: Autoria própria.**

A macro que contém todos os módulos e os formulários foi disponibilizada no endereço eletrônico: “<https://goo.gl/ACZ0nD>”. Desta forma, é possível executar o programa e verificar toda a codificação desenvolvida no projeto.

### 5.3.5 Criação dos formulários de interação com o usuário

Foram criados formulários no VBA no intuito de permitir a interação com o usuário. Com isso, é possível pegar os dados que são passados aos campos gerados no formulário, converter o que for necessário e passá-los às variáveis utilizadas na codificação, conforme já foi citado anteriormente. Para permitir a inserção dos dados pelo usuário foram utilizados campos de texto e para iniciar a codificação ou fechar o formulário foram adicionados botões. Também foram adicionados rótulos ao formulário, responsáveis por identificar os campos e duas imagens, uma contendo a explicação de quais são as medidas alteradas por determinado campo do formulário e outra que mostra uma ideia de como ficará o exaustor depois de realizada a execução.

Foi criado, também, um formulário de apresentação, que é exibido precedendo o formulário principal de definição dos dados. Este contém imagens do logo criado para o programa, logo da UTFPR e logo da Fundação CAPES, além de rótulos responsáveis por trazer informações referentes ao programa. A Figura 36 e a Figura 37 apresentam o formulário de apresentação e o formulário de definição de dados, respectivamente.



**Figura 36 - Formulário de apresentação.**  
**Fonte: Autoria própria.**

SolidEx API ✕

Dados de entrada

Nome do projetista:

Nome da empresa:

Largura/Altura da Carçaça:  Deve estar entre 1200 mm e 3000 mm

Profundidade da Carçaça:  Deve ser maior que 200 mm e ser menor que a Largura/Altura da Carçaça

Profundidade do Cone:  Deve ser maior que 120 mm e menor que o triplo da Profundidade da Carçaça

Espessura do Material:  Deve ser entre 1 mm e 3 mm

Número de Hélices:  Deve ser entre 2 e 8

Diagrama de uma hélice com dimensões indicadas: Profundidade do Cone, Profundidade da Carçaça e Largura/Altura da Carçaça.

Imagem 3D de um ventilador com uma grade de proteção.

Pasta de Destino:  ...

Formato da Folha:  ...

**Figura 37 - Formulário de inserção de dados.**  
**Fonte: Autoria própria.**

## 6 CONCLUSÕES E TRABALHOS FUTUROS

Nesta seção, serão abordados aspectos relativos às conclusões acerca do trabalho desenvolvido e sugestões para abordagens futuras.

### 6.1 CONCLUSÕES

Ao final do trabalho, tornou-se claro o quão abrangente é a área possível de ser trabalhada utilizando a API do SolidWorks® na automação de projetos, uma vez que qualquer ação desenvolvida diretamente dentro do *software* pode ser executada, também, através da codificação de macros no Visual Basic for Applications®.

Por ser uma área bastante ampla e pouco explorada, a automação de projetos na indústria metal-mecânica, voltada para a manufatura de equipamentos agroindustriais, é um assunto que pode ser muito mais explorado, principalmente na tentativa de colaborar com empresas de pequeno e médio porte que normalmente não dispõem de um projetista especializado na empresa e acabam ficando sem o acesso a este importante recurso.

Devido ao fato de que, na maioria das vezes, o projetista que trabalha com o SolidWorks® não ser um profissional da área da computação, a programação se torna um empecilho considerável na utilização de macros e conseqüentemente na automação destes projetos. Sabendo disso, publicou-se um livro contendo, passo a passo, cada um dos métodos, propriedades e interfaces utilizadas na execução deste trabalho. Com isso, até mesmo projetistas leigos em programação, ou estudantes realizando pesquisas nesta mesma área, podem verificar todos os aspectos trabalhados aqui e, além de conseguir desenvolver este mesmo projeto, fazer qualquer outro tipo de automação dentro do *software*. O endereço eletrônico para acesso deste material é: "<https://goo.gl/9IKCuN>".

## 6.2 TRABALHOS FUTUROS

No presente trabalho podem ser trabalhados projetos de extensão, como a criação de uma plataforma responsável por sugerir e automatizar a decisão quanto a quantidade de unidades e dimensionamento dos exaustores a serem projetados, no qual seja desenvolvido um formulário em que o usuário possa inserir dados referentes às dimensões do barracão e este calcule, utilizando estes dados, a quantidade e tamanho adequados de exaustores para este determinado espaço.

Pode-se automatizar o projeto de parafusos com diversos tamanhos, diâmetros, roscas e tipos de cabeça diferentes, de modo que o usuário selecione o tipo de cabeça dentre algumas determinadas opções e defina as medidas de comprimento, diâmetro, passo e tipo de rosca que deseja e a macro desenvolva o projeto desta peça automaticamente.

Outro exemplo é a inserção de informativos de custo e quantidade de material a ser utilizada, trazendo diferentes opções de material a serem aplicados na fabricação e o quanto esta decisão impacta na construção do equipamento relativo a estes dois aspectos.

Pode-se pensar, também, em alternativas no ambiente de programação, uma vez que o SolidWorks® disponibiliza ao programador trabalhar com outras linguagens de programação diferentes da Visual Basic. Linguagens de programação totalmente orientadas à objetos, como é o caso da linguagem VB.NET, podem trazer recursos interessantes e possibilidades de vinculação adicionais ao usuário, como acesso à banco de dados externo ou criação de interfaces gráficas mais elaboradas.

É possível, também, fazer alterações no atual projeto a fim de torná-lo mais completo na automação da criação de projetos de exaustores, uma vez que até o momento não é possível determinar informações quanto ao formato de pá ou sistema de tração das hélices, por exemplo.

## REFERÊNCIAS

AGÊNCIA EMBRAPA DE INFORMAÇÃO TECNOLÓGICA. Disponível em: <[http://www.agencia.cnptia.embrapa.br/gestor/frango\\_de\\_corte/arvore/CONT000fzfgnol202wx5ok0cpoo6aqwp0qni.html](http://www.agencia.cnptia.embrapa.br/gestor/frango_de_corte/arvore/CONT000fzfgnol202wx5ok0cpoo6aqwp0qni.html)>. Acesso em: 10 set. 2015.

ALI, Hazrat, KATSUKI, Akio, KUROKAWA, Syuhei, SAJIMA, Takao. Design of mechanical actuator in 3D CAD software. **Procedia Engineering**, v. 64, p. 473–478, 2013. Elsevier B.V. Disponível em: <<http://dx.doi.org/10.1016/j.proeng.2013.09.121>>. Acesso em: 10 set. 2015.

ASSOCIAÇÃO BRASILEIRA DE PROTEÍNA ANIMAL. **Relatório Anual**. Relatório anual de atividades 2014. São Paulo, 2014.

AVIOESTE, Corti. Exaustor EOC 53” – Dados técnicos. Disponível em: <<http://www.avioeste.com.br/produtos/20/exaustor-eoc-53>>. Acesso em: 15 nov. 2016.

BELUSSO, Diane. A evolução da avicultura industrial brasileira e seus efeitos territoriais. **Revista Percursos – NEMO**, v.2, n. 1, p. 25–51, 2010. Disponível em: <<http://www2.fct.unesp.br/nivaldo/Publica%E7%F5es-nivaldo/2010/AVICULTURA-2010.pdf>>. Acesso em: 05 mai. 2015.

BUENO, Leda; ROSSI, Luiz A. Comparação entre tecnologias de climatização para criação de frangos quanto a energia, ambiência e produtividade. **Revista Brasileira de Engenharia Agrícola e Ambiental**, v. 10, n. 2, p. 497–504, 2006. Disponível em: <<http://www.scielo.br/pdf/rbeaa/v10n2/v10n2a35.pdf>>. Acesso em: 08 set. 2015.

CARDOSO, Andréa. **Fundamentos da pesquisa operacional**. Alfenas, Minas Gerais, 2011. 102 p.

CARNIATO, Antônio.; BETTIOL, Arlan L.; UGGIONI, Gilberto D.; BETTIOL,Áureo D.; PASSOS, Luis F. do N.; PENNER, Norbert; ROMANCINI, José A.; RODRIGUES, Alexandre C.; DA SILVA, Wladimir A. **Eficiência Energética em Aviários : Geração de Energia Elétrica e Sistema de Armazenamento para Faltas**. VIII Congresso de Inovação Tecnológica em Energia Elétrica. Mata de São João, Bahia. 2015.

CELIK, H. Kursat; RENNIE, Allan E. W.; AKINCI, Ibrahim. Deformation behaviour simulation of an apple under drop case by finite element method. **Journal of Food Engineering**, v. 104, n. 2, p. 293–298, 2011. Elsevier Ltd. Disponível em: <<http://dx.doi.org/10.1016/j.jfoodeng.2010.12.020>>. Acesso em: 08 set. 2015.

CHAKRAVORTY, Tulika; CHAKRABORTY, Samyadip; JIGEESH, Nasina. Analysis of Agile Testing Attributes for Faster Time to Market: Context of Manufacturing Sector Related IT Projects. **Procedia Economics and Finance**, v. 11, n. 14, p. 536–552, 2014. Elsevier B.V. Disponível em: <<http://www.sciencedirect.com/science/article/pii/S2212567114002196>>. Acesso em: 11 set. 2015.

CHERNAKI-LEFFER, Andreia. M., BIESDORF, Sônia M., ALMEIDA, Lucia M., LEFFER, Eduardo V. B., VIGNE, F. Isolamento de enterobactérias em *Alphitobius diaperinus* e na cama de aviários no oeste do estado do Paraná, Brasil. **Revista Brasileira de Ciência Avícola**, v. 4, n. 3, 2002. Disponível em: <<http://www.scielo.br/pdf/rbca/v4n3/14670.pdf>>. Acesso em: 10 set. 2015.

CHOI, Guk-Heon; MUN, Duhwan; HAN, Soonhung. Exchange of CAD part models based on the macro-parametric approach. **International Journal of CAD/CAM**, v. 2, n. 1, p. 13–21, 2009. Disponível em: <<http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.130.6566&rep=rep1&type=pdf>><<http://www.ijcc.org/ojs/index.php/ijcc/article/view/8>>. Acesso em: 09 set. 2015.

CREMONEZ, Paulo A.; FEROLDI, Michael; NADALETI, Willian C.; DE ROSSI, Eduardo; FEIDEN, Armin; DE CAMARGO, Mariele P.; CREMONEZ, Filipe E.; KLAJN, Felipe F. Biodiesel production in Brazil: Current scenario and perspectives. **Renewable and Sustainable Energy Reviews**, v. 42, p. 415–428, 2015. Elsevier Ltd. Disponível em: <<http://www.scopus.com/inward/record.url?eid=2-s2.0-84908399134&partnerID=tZOtx3y1>>. Acesso em: 01 mar. 2015.

FARHAN, Uday H.; RAD, Majid T. Design of modular fixtures using a 3D-modelling approach. **19<sup>th</sup> International Congress on Modelling and Simulation**, Perth, Australia, 12-16 Dez. 2011.

FARHAN, Uday H.; O'BRIEN, Simona; RAD, Majid T. SolidWorks Secondary Development with Visual Basic 6 for an Automated Modular Fixture Assembly Approach. **International Journal of Engineering**, v. 6, p. 290–304, 2012.

FUNCK, Sandro R.; FONSECA, Ricardo A. Avaliação energética e de desempenho de frangos com aquecimento automático a gás e a lenha. **Revista Brasileira de Engenharia Agrícola e Ambiental**, v. 12, n. 1, p. 91–97, 2008. Disponível em: <<http://www.scielo.br/pdf/rbeaa/v12n1/v12n01a14.pdf>>. Acesso em: 10 set. 2015.

GAGNEZ, E. T. Solidworks premium. Design without limits. 2013. 8 p.

GAO, Peng; SANG, Zhi-fu. Three Dimensional Parameterized Design of Heat Exchanger Tubesheet Based on SolidWorks. **Computer-Aided Industrial Design and Conceptual Design**. p. 488–492, 22-25 nov. 2008.

GIL, Antonio C. **Como elaborar projetos de pesquisa**. 5. ed. São Paulo, 2010. 184 p.

GROENER, Brian; KNOPP, Nile; KORGAN, Kent; PERRY, Rowen; ROMERO, Jonathan; SMITH, Kody; STAINBACK, Albert; STRZELCZYK, Austin; HENRIQUES, Justin. Preliminary Design of a Low-cost Greenhouse with Open Source Control Systems. **Procedia Engineering**, v. 107, p. 470–479, 2015. Elsevier B.V. Disponível em: <<http://linkinghub.elsevier.com/retrieve/pii/S1877705815010589>>. Acesso em: 09 set. 2015.

KAUARK, Fabiana da S.; MANHÃES, Fernanda C.; MEDEIROS, Carlos H. **Metodologia da pesquisa: Um guia prático**. Itabuna, Bahia, 2010. 89 p.

KONG, L.; FUH, J. Y. H.; LEE, K. S.; LIU, X. L.; LING, L. S.; ZHANG, Y. F.; NEE, A. Y. C. **Journal of Materials Processing Technology**, v. 139, n. 1-3 SPEC, p. 81–89, 2003. Disponível em: <[http://ac.els-cdn.com/S0924013603001869/1-s2.0-S0924013603001869-main.pdf?\\_tid=e6bf07f6-607d-11e5-93f7-00000aab0f26&acdnat=1442853136\\_09e353dcfe52620967e594374f2dd0e2](http://ac.els-cdn.com/S0924013603001869/1-s2.0-S0924013603001869-main.pdf?_tid=e6bf07f6-607d-11e5-93f7-00000aab0f26&acdnat=1442853136_09e353dcfe52620967e594374f2dd0e2)>. Acesso em: 10 set. 2015.

MATHEW, Arun T. A Novel Method of Using API to Generate Liaison Relationships from an Assembly. **Journal of Software Engineering and Applications**, v. 03, n. 02, p. 167–175, 2010. Disponível em: <<http://file.scirp.org/Html/1361.html>>. Acesso em: 10 set. 2015.

MENDA, František; ŠARGA, Patrik; PRADA, Erik; TREBUŇA, František. SolidWorks API for Ring-Core simulations. **IEEE 12<sup>th</sup> International Symposium on Applied Machine Intelligence and Informatics**. SAMI 2014, p. 151-155. 2014, Herl'any, Slovakia. Disponível em: <<http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=6822397>>. Acesso em: 18 nov. 2015.

MICROSOFT. **Visão geral sobre requisitos e conceitos OLE**. Disponível em: <<https://support.microsoft.com/pt-br/kb/86008>>. Acesso em 12 nov. 2015.

MICROSOFT CORPORATION. **Microsoft Visual Basic 6.0: Programmer's Guide**. 1. ed. Redmond, WA, USA: Microsoft Press, 1998. 960 p.

MICROSOFT OFFICE PROFESSIONAL PLUS 2013. Microsoft Corporation. Imprensa Systems, Santa Rosa, California.

MSD Escola Aberta. **Curso de Visual Basic 6.0**. Brasília, Distrito Federal. MSD Editora, 1999. 200 p.

NING, Xiao B.; JIANG, Quan S. A digital design method of geometric model for centrifugal fan impeller based on solidworks and VB. **Proceedings of 2011 International Conference on Electronic and Mechanical Engineering and Information Technology, EMEIT 2011**, v. 8, p. 4023–4026, 2011. Disponível em: <<http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=6023935>>. Acesso em: 11 set. 2015.

OLIVEIRA, Antonio C. de. **Diretrizes de apoio ao esforço de inovação tecnológica no desenvolvimento de produtos em pequenas e médias empresas industriais**. 2007. 218 f. Tese (Doutorado em Engenharia) – Escola Politécnica da Universidade de São Paulo, Universidade de São Paulo, São Paulo, 2008.

PRINCE, Stewart P.; RYAN, Robert G.; MINCER, Tom. Common API: Using Visual Basic to Communicate between Engineering Design and Analytical Software Tools. **Journal of Chemical Information and Modeling**, v. 53, p. 1689–1699, 2013.

PERRY, Greg M. **Aprenda em 21 dias Visual Basic 6**. Gulf Professional Publishing, 1999. 844 p.

RODRIGUES, Wesley O. P., GARCIA, Rodrigo G., NÄÄS, Irenilza de A., DA ROSA, Carolina O., CALDARELLI, Carlos E. Evolução da avicultura de corte no Brasil. **Enciclopédia Biosfera**, v. 10, n. 18, p. 1666–1684, 2014. Disponível em: <<http://www.conhecer.org.br/enciclop/2014a/AGRARIAS/EVOLUCAO.pdf>>. Acesso em: 09 set. 2015.

SILVA, Edna L. da; MENEZES, Estera M. **Metodologia da pesquisa e elaboração de dissertação**. 4. ed. Florianópolis, 2005. 123 p.

SOLIDWORKS 2013. Edição Premium x64, Versão SP0.0. Delaware, Estados Unidos da América: Dassault Systèmes SolidWorks Corporation. 1 DVD-ROM.

SPENS, Mike. **Automating SOLIDWORKS 2015 Using Macros**, SDC Publications, 2015. 384 p.

SUN, Bo; QIN, Guangtai; FANG, Yadong. Research of standard parts library construction for SolidWorks by Visual Basic. **Proceedings of 2011 International Conference on Electronic and Mechanical Engineering and Information Technology**, EMEIT 2011, v. 5, p. 2651–2654, 2011. Disponível em: <<http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=6023642>>. Acesso em: 09 set. 2015.

TINOCO, Hector A., OCAMPO, Diego A., PEÑA, Fabio M., SANZ-URIBE, Juan R. Finite element modal analysis of the fruit-peduncle of *Coffea arabica* L. var. Colombia estimating its geometrical and mechanical properties. **Computers and Electronics in Agriculture**, v. 108, p. 17–27, 2014. Elsevier B.V. Disponível em: <<http://linkinghub.elsevier.com/retrieve/pii/S0168169914001719>>. Acesso em: 10 set. 2015.

TITUS, Avitus; BIN, Liu X. Secondary Development of Solid works for Standard Components Based on Database. **International Journal of Science and Research**, v. 2, n. 10, p. 162–164, 2013.

TONIN, Paulo C. **SolidWorks: Montagem e detalhamento básicos**. 1. ed. Medianeira, Paraná, 2015. 163 p.

YONG, Wu G.; XING, Tang G.; LIANG, Zhang H.; SHAN, Yao. Visual Modeling and Simulation for Laser Action Area in Rapid Prototyping. **2009 WRI World Congress on Computer Science and Information Engineering**, p. 435–439, 2009. Disponível em: <<http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=5170573>>. Acesso em: 12 nov. 2015.