

UNIVERSIDADE PAULISTA – UNIP

PROGRAMA COMPUTACIONAL PARA O CÁLCULO DE
ÍNDICES DE CONFORTO TÉRMICO NA PRODUÇÃO
INDUSTRIAL DE ANIMAIS PARA CARNE E LEITE

MARIANO SERGIO PACHECO DE ANGELO

SÃO PAULO
2013

MARIANO SERGIO PACHECO DE ANGELO

PROGRAMA COMPUTACIONAL PARA O CÁLCULO DE ÍNDICES DE
CONFORTO TÉRMICO NA PRODUÇÃO INDUSTRIAL DE ANIMAIS
PARA CARNE E LEITE

Dissertação apresentada ao Programa de Pós-Graduação em Engenharia de Produção da Universidade Paulista – UNIP para a obtenção do título de Mestre em Engenharia de Produção

Orientador: Prof. Dr. Oduvaldo Vendrametto

SÃO PAULO

2013

Angelo, Mariano Sergio Pacheco de.

Programa computacional para o cálculo de índices de conforto térmico na produção industrial de animais para carne e leite / Mariano Sergio Pacheco de Angelo - 2013.

66 f. : il. color. + DVD.

Dissertação de mestrado apresentada ao Programa de Pós-Graduação em Engenharia da Produção da Universidade Paulista, São Paulo, 2013.

Área de Concentração: Gestão de Sistemas de Operação.

Orientador: Prof. Dr. Oduvaldo Vendrametto.

MARIANO SERGIO PACHECO DE ANGELO

PROGRAMA COMPUTACIONAL PARA O CÁLCULO DE ÍNDICES DE CONFORTO
TÉRMICO NA PRODUÇÃO INDUSTRIAL DE ANIMAIS PARA CARNE E LEITE

Dissertação apresentada ao Programa de Pós-
Graduação em Engenharia de Produção da
Universidade Paulista – UNIP para a obtenção
do título de Mestre em Engenharia de Produção

Aprovado em ____/____/____

Banca Examinadora

Profa. Dra. Irenilza Nääs (UNIP)

Profa. Dra. Marta Baracho (UNICAMP)

Prof. Dr. Oduvaldo Vendrametto (UNIP)

DEDICATÓRIA

Aos meus pais,

Daniel de Angelo (*in memoriam*)

Esther Pacheco de Angelo

Pelo incentivo, paciência, compreensão e pelas lições ministradas através dos exemplos de vida, com que sempre me brindaram ao longo de todos esses anos. Que Deus os abençoe!

AGRADECIMENTOS

Agradeço imensamente aos professores Oduvaldo Vendrametto e Irenilza Nääs, por terem sido amigos, companheiros, mestres e principalmente, por terem acreditado em mim, até mesmo quando eu já havia perdido a fé.

A Bárbara Gutierrez, Fábio Vieira do Amaral, Mario Quinello e Marcelo Nogueira, sem eles eu não teria conseguido finalizar esta empreitada.

Aos professores Pedro Luiz, Mario Mollo e Ivanir Ribeiro, pela paciência e pelos conhecimentos que me foram transmitidos.

A todos os meus colegas de curso, pela convivência que me proporcionaram durante esse período. A troca de experiências ficará marcada para sempre em meus pensamentos.

A todo o pessoal da secretaria, sem os quais eu jamais teria conseguido superar os trâmites burocráticos inerentes ao processo.

Ao CAPES, que através do Programa de Suporte à Pós-Graduação de Instituições de Ensino Particulares (PROSUP), tornou possível a realização desta empreitada ao me conceder uma bolsa de estudo.

“Embora ninguém possa voltar atrás e fazer um novo
começo, qualquer um pode começar agora a fazer
um novo futuro.”

Francisco Cândido Xavier

RESUMO

O Brasil é o maior exportador mundial de carnes bovina e de frango, ocupando a terceira posição na exportação de carne suína. Incrementos na produtividade geram maiores lucros para toda a cadeia de produção. Uma das causas do sucesso de um empreendimento para a produção animal está relacionada ao ambiente a que estarão submetidos os indivíduos alojados. O ambiente térmico engloba os efeitos da temperatura ambiente, da radiação solar, da umidade relativa e velocidade do ar. O objetivo deste trabalho foi desenvolver um programa computacional que identificasse situações ótimas de alojamento em relação ao conforto térmico, através de poucas variáveis de fácil obtenção no ambiente de produção industrial para frangos, suínos e bovinos de leite, sugerindo ações ao produtor para a mitigação dos riscos. A metodologia foi desenvolvida em duas partes: foram consultados na literatura os índices relacionados ao conforto térmico na produção industrial de animais; a partir dos algoritmos identificados foram aplicados os paradigmas da engenharia de software no desenvolvimento do programa computacional. Mediante os resultados obtidos pela metodologia aplicada, foi possível desenvolver um programa computacional para calcular os índices de conforto térmico para frangos, suínos e bovinos de leite em ambientes de produção intensiva, permitindo um maior controle do ambiente da produção.

Palavras-Chave: Conforto Térmico, Engenharia de *Software*, Produção Industrial de Animais

ABSTRACT

Brazil is the largest exporter of beef and chicken meat, ranking third in the export of pork. Increases in productivity generates higher profits for the entire production chain. One of the causes of the success of an enterprise for animal production is related to the environment that will be submitted housed individuals. The thermal environment encompasses the effects of temperature, solar radiation, relative humidity and air velocity. The aim of this study was to develop a computer program to identify situations of great accommodation in relation to thermal comfort through few variables easily obtained in industrial production environment for chickens, pigs and dairy cattle, suggesting actions to the producer for the mitigation of risks. The methodology was developed in two parts, were found in the literature related to thermal comfort indices in industrial production of animals; identified from the algorithms were applied to the paradigms of software engineering in the development of the software. From the results obtained by the methodology applied, it was possible to develop a software to calculate the thermal comfort index for chickens, pigs and cattle for milk production intensive environments, allowing greater control over the production environment

Keywords: Thermal Comfort, Software Engineering, Industrial Production Animal

LISTA DE SIGLAS E ABREVIATURAS

ASP	Active Server Pages
CASE	Computer Aided Software Engineering
CMM	Capability Maturity Model
COCOMO	Constructive Cost Model
ERP	Enterprise Resource Planning
ES	Engenharia de Software
IGNU	Índice de Globo Negro e Umidade
ITU	Índice de Temperatura e Umidade
ITUV	Índice de Temperatura Umidade e Velocidade
OMG	Object Management Group
OO	Orientação a Objetos
OOA	Object Oriented Analysis
OOAD	Object Oriented Analysis and Design
OOD	Object Oriented Design
RAD	Rapid Application Development
RUP	Rational Unified Process
SEI	Software Engineering Institute
SQA	Software Quality Assurance
UM	Unified Method
UML	Unified Modeling Language

LISTA DE FIGURAS

Figura 1 – Imagem termográfica infravermelha de bovinos de leite	16
Figura 2 - Alojamento de frangos de corte	17
Figura 3 - Imagem termográfica do alojamento de frangos de corte	17
Figura 4 - Efeito da ventilação sob a temperatura superficial de porcas em lactação	17
Figura 5 - Variação da Produção de calor em Homeotermos em Função da Temperatura Ambiente	18
Figura 6 - Ciclo de vida clássico do software	25
Figura 7 - Combinando Paradigmas da ES	25
Figura 8 - Exemplo de um Diagrama de Classe da UML	35
Figura 9 - Exemplo de um Diagrama de Caso de Uso da UML	35
Figura 10 - Exemplo de um Diagrama de Sequência da UML	36
Figura 11- Fluxograma dos procedimentos para a execução da presente pesquisa	39
Figura 12- Exemplo de Diagrama de Caso de Uso utilizado no programa computacional	40
Figura 13- Exemplo de Diagrama de Sequência utilizado no programa computacional	40
Figura 14 - Exemplo de Diagrama de Classes utilizado no programa computacional	41
Figura 15 - Interface Gráfica com o Usuário	42
Figura 16 - Tela exemplo do cálculo do ITU para aves de corte	44
Figura 17 - Tela exemplo do cálculo do ITGU para bovinos	44
Figura 18 - Tela exemplo do cálculo do ITU para suínos	45
Figura 19 - Dados obtidos para o cálculo do ITU dentro da faixa de normalidade	53
Figura 20 - E-mail enviado pelo sistema ao atingir o ITU dentro da faixa de conforto	53
Figura 21 - Dados obtidos para o cálculo do ITU dentro da faixa de atenção	54
Figura 22 - E-mail enviado pelo sistema ao atingir o ITU dentro da faixa de atenção	54
Figura 23 - Retorno ao estado de conforto devido ao acionamento dos equipamentos (Alerta)	55
Figura 24- Dados obtidos para o cálculo do ITU dentro da faixa de PERIGO	56
Figura 25 - E-mail enviado pelo sistema ao atingir o ITU dentro da faixa de Perigo	56
Figura 26 - Retorno ao estado de conforto devido ao acionamento dos equipamentos (Perigo)	57
Figura 27 - Dados obtidos para o cálculo do ITU dentro da faixa de EMERGÊNCIA	58
Figura 28- E-mail enviado pelo sistema ao atingir o ITU dentro da faixa de emergência	58
Figura 29 - Retorno ao estado de conforto devido ao acionamento dos equipamentos (Emergência)	59

LISTA DE TABELAS

Tabela 1 - Valores Médios de temperatura retal (°C)	18
Tabela 2 - Fonte dos algoritmos para determinação dos Índices de Conforto Térmico.	37
Tabela 3 - Algoritmos para determinação dos Índices de Conforto Térmico	38
Tabela 4 - Valores limites de ITU, ITUV e IGNU	38
Tabela 5 – ITU - Valores e Status obtidos através de simulação utilizando dados hipotéticos a partir de uma série de dados aleatórios (Frangos).....	50
Tabela 6 - ITUV - Valores e Status obtidos através de simulação utilizando dados hipotéticos a partir de uma série de dados aleatórios (Frangos).....	51
Tabela 7 - IGNU - Valores e Status obtidos através de simulação utilizando dados hipotéticos a partir de uma série de dados aleatórios (Frangos).....	52

SUMÁRIO

1 CAPÍTULO I - INTRODUÇÃO.....	14
1.1 Objetivo Geral.....	15
1.1.1 Objetivos Específicos.....	15
2 CAPÍTULO II - REVISÃO BIBLIOGRÁFICA	16
2.1 Produção Intensiva de Animais	16
2.2 Índices de Conforto Térmico na Produção Animal	19
2.2.1 Índice de Temperatura e Umidade (ITU)	20
2.2.2 Índice de Temperatura Umidade e Velocidade (ITUV)	21
2.2.3 Índice de Globo Negro e Umidade (IGNU)	22
2.3 Engenharia de Software	23
2.3.1 Paradigmas da Engenharia de Software.....	23
2.3.2 Gerenciamento de Projetos	26
2.3.3 Requisitos e Análise	27
2.3.4 Projeto e Implementação	28
2.3.5 Testes, Manutenção e Qualidade de Software	29
2.3.6 Orientação a Objetos	31
2.3.7 Unified Modeling Language (UML)	33
3 CAPÍTULO III - METODOLOGIA.....	37
3.1 Organização dos Índices da Literatura	37
3.2 Fluxo dos Processos para Definição do Programa Computacional.....	39
3.3 Diagramas UML	39
3.3.1 Caso de Uso	39
3.3.2 Sequência	40
3.3.3 Classes	41
3.4 Desenvolvimento do Programa Computacional.....	41
3.4.1 Interface Gráfica com o Usuário.....	42
3.4.2 Codificação dos Algoritmos	45
3.4.3 Interface com Equipamentos de Coletas de Dados	45
3.4.4 Envio de e-mails.....	46
3.4.5 MultiThread	46
3.4.6 Ciclos de Desenvolvimento	46
4 CAPÍTULO IV - ANÁLISE E DISCUSSÃO DOS RESULTADOS.....	48
4.1 Intervalos Utilizados no Programa Computacional	48
4.2 Validação dos Resultados Obtidos	49
4.2.1 ITU	49

4.2.2 ITUV.....	51
4.2.3 IGNU.....	52
4.3 Exemplos da Utilização do Programa Computacional	52
5 CAPÍTULO V - CONCLUSÃO	60
5.1 Propostas para Estudos Futuros.....	61
REFERÊNCIAS BIBLIOGRÁFICAS	62

CAPÍTULO I

1 INTRODUÇÃO

O Ministério da Agricultura estima que até 2020, a produção nacional de carnes suprirá 44,5% do mercado mundial, a carne de frango representará 48,1% das exportações mundiais, a participação da carne suína será de 14,2%. O Brasil é atualmente o maior exportador mundial de carnes bovina e de frango, ocupando a terceira posição na exportação de carne suína.

Uma das principais causas do sucesso de um empreendimento para a produção animal está relacionada ao ambiente a que estarão submetidos os indivíduos alojados (PEIXOTO, 2004). O ambiente térmico engloba os efeitos da temperatura ambiente, da radiação solar, da umidade relativa do ar e da velocidade do ar. A combinação da umidade relativa do ar e da temperatura ambiente é o principal condicionante para o conforto térmico e o funcionamento geral dos processos fisiológicos dos animais (OLIVEIRA, 1995; FALCO, 1997; BAÊTA e SOUZA, 1997). Tomando-se, como exemplo, a área da avicultura, em que a dificuldade apresentada pelas aves na troca térmica com o ambiente é fortemente afetada pelas instalações, que, quando não promovem o equilíbrio térmico desejável, fazem com que as amplitudes críticas externas sejam transferidas para o interior dos galpões, provocando altos índices de mortalidade (NÃÃS *et al.*, 1995) e influenciando negativamente índices de produtividade.

Para a criação de animais de produção de carne, leite e ovos nas condições brasileiras de temperatura e umidade, a cobertura ideal das instalações deve ter considerável capacidade de isolamento térmico, possuir uma boa capacidade de retardo térmico, ser resistente às intempéries, de fácil manuseio e montagem, resistente mecanicamente, ter bom aspecto estético, ser de baixo custo, impermeável, leve e apresentar grande capacidade para refletir a radiação solar (BAÊTA e SOUZA, 1997; MORAES, 1999). Nessas condições, o sombreamento proveniente de coberturas bem dimensionadas, reduz entre 20% e 40% da radiação no interior de instalações para animais (SANTOS *et al.*, 1993; TURCO *et al.*, 1994; ABREU *et al.*, 1995; BAÊTA; SOUZA, 1997).

A diversidade de níveis culturais dos produtores no Brasil é muito grande, e muitos não têm acesso às modernas técnicas de produção animal. O objetivo deste

trabalho foi desenvolver um programa computacional que permita o cálculo de índices de conforto para animais alojados em condições industriais de produção, através da digitação de alguns dados de fácil obtenção pelo produtor. Sugerindo em um primeiro momento algumas ações simples que possam colaborar para a mitigação dos riscos envolvidos pelo desequilíbrio dos fatores térmicos, ajudando ainda no aumento da produtividade. Esses objetivos são detalhados no Capítulo I.

No Capítulo II, foi realizada uma revisão bibliográfica para contextualizar o escopo deste trabalho quanto aos índices de conforto aqui utilizados e as principais definições da Engenharia de Software, que é a base para o programa computacional desenvolvido no decorrer do trabalho.

O detalhamento da metodologia encontra-se no Capítulo III.

No Capítulo IV os resultados são apresentados e discutidos.

As conclusões são apresentadas no Capítulo V, juntamente com sugestões para continuação das pesquisas.

Ao final são listadas as referências bibliográficas.

1.1 Objetivo Geral

Desenvolver um programa computacional que possa identificar situações ótimas de alojamento em relação ao conforto térmico, através de poucas variáveis de fácil obtenção no ambiente de produção industrial para frangos, suínos e bovinos de leite, sugerindo ações ao produtor para a mitigação dos riscos.

1.1.1 Objetivos Específicos

1. Calcular os índices de conforto térmico para frangos, suínos e bovinos de leite em ambientes de produção intensiva.
2. Identificar os intervalos dos limites de conforto térmico dos alojamentos relacionados à produção.
3. Aplicar os paradigmas da Engenharia de software no desenvolvimento de um programa computacional que implemente os itens 1 e 2.
4. Validar o programa computacional através de valores descritos na literatura e entrevistas com especialistas da área.

CAPÍTULO II

2 REVISÃO BIBLIOGRÁFICA

2.1 Produção Intensiva de Animais

Em condições de ambientes quentes, como são as várias regiões brasileiras, o excesso de calor pode prejudicar a produção intensiva de animais (frangos, suínos e bovinos de leite).

Animais homeotermos requerem condições de alojamento que possibilitem um permanente equilíbrio térmico, de forma que a energia consumida através da ração resulte em produção (carne, leite e ovos). Quando a temperatura do alojamento está abaixo da zona de termo neutralidade, que é a ideal, os animais sofrem de frio. Por outro lado, quando a temperatura ambiente se encontra acima da ideal, os animais sofrem de calor.

Como exemplo, mostram-se nas Figuras 1, 2, 3 e 4 imagens de animais alojados em condições intensivas de produção, nas quais se pode identificar a fonte de calor.

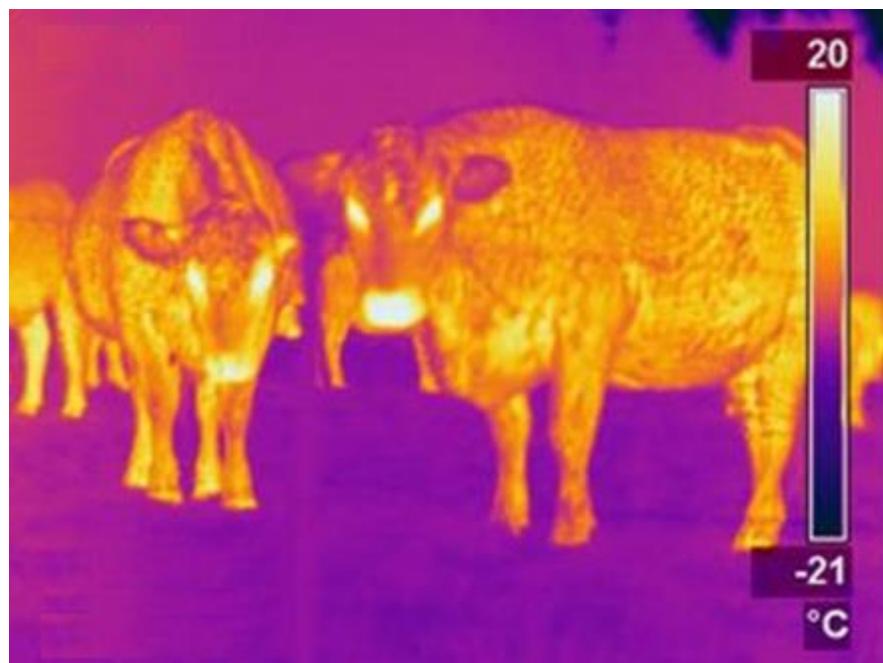


Figura 1 – Imagem termográfica infravermelha de bovinos de leite



Figura 2 - Alojamento de frangos de corte

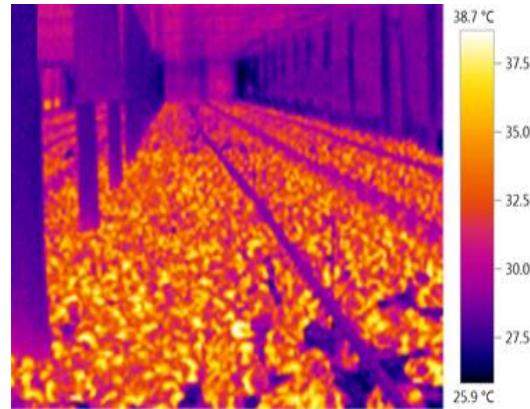


Figura 3 - Imagem termográfica do alojamento de frangos de corte

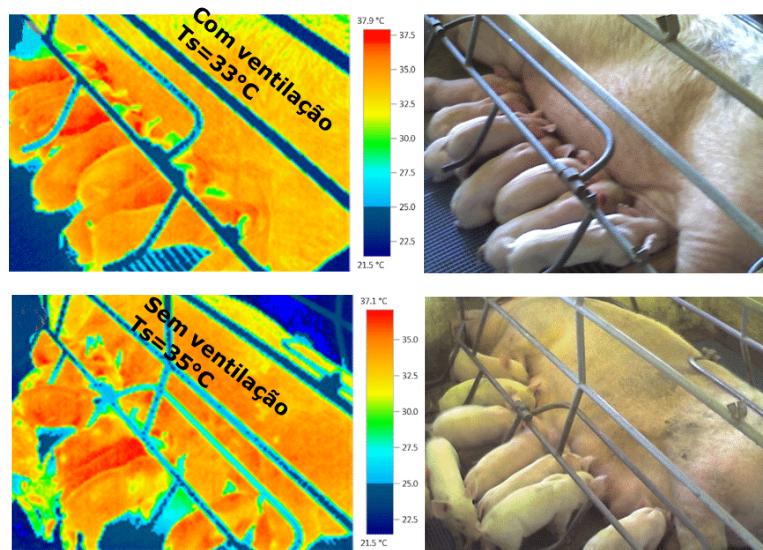


Figura 4 - Efeito da ventilação sob a temperatura superficial de porcas em lactação

Para realizar o balanço de energia, o animal reage às condições do ambiente térmico fazendo ajustes fisiológicos e comportamentais. Os comportamentais permitem ao animal a possibilidade de alterar seu microclima, enquanto os fisiológicos buscam a adaptação. Ambos objetivam o conforto térmico e sugerem que o animal é o melhor indicador das condições térmicas do ambiente.

A manutenção da temperatura corporal (Tabela 1), sob condições ambientais termicamente estressantes, requer alteração de várias funções do animal, em razão do fluxo de calor que se estabelece entre o animal e o ambiente: nas condições de altas temperaturas ambientais, o animal necessita reduzir sua taxa de produção e conservação de calor e incrementar a dissipação de calor corporal para o ambiente, para evitar a elevação de sua temperatura corporal; nas condições de baixas

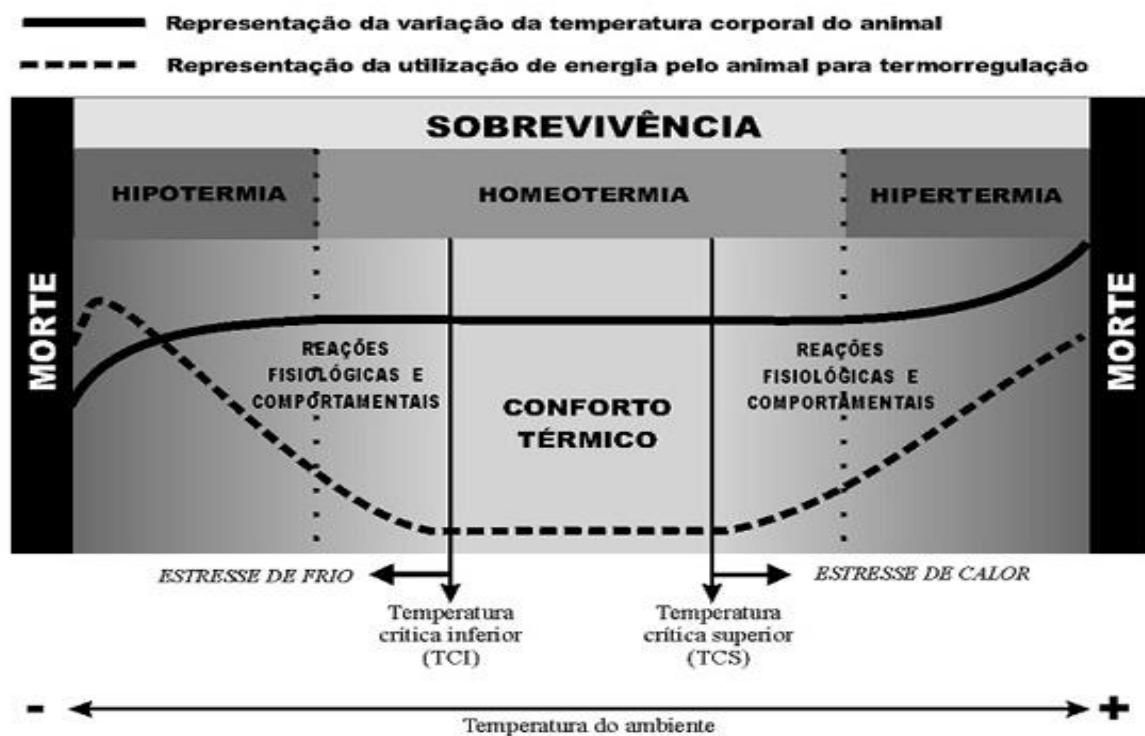
temperaturas ambientais, os processos de conservação e produção de calor corporal precisam ser intensificados, para compensar o incremento na dissipação desse calor, em razão da maior demanda térmica do ambiente.

Tabela 1 - Valores Médios de temperatura retal (°C)

Animal	Temperatura Média
Bovinos (corte)	38,3
Bovinos (leite)	38,6
Suínos	39,2
Frangos	41,0

Fonte: Baeta e Souza, 1997.

A termo regulação em animais homeotermos ocorre por meio de mecanismos comportamentais e fisiológicos. Esse comportamento característico está expresso na Figura 5.



Na zona Termo neutra, acontece o intervalo de temperaturas em que o animal não precisa produzir ou perder calor para o meio para manter a sua temperatura corporal, o metabolismo é mínimo e o desempenho otimizado.

Abaixo da temperatura crítica inferior (TCI), o animal aciona seus mecanismos termo regulatórios para INCREMENTAR a produção e RETENÇÃO de calor corporal, compensando a perda de calor para o ambiente, que se encontra frio.

Acima da temperatura crítica superior (TCS), o animal aciona seus mecanismos termo regulatórios para auxiliar a DISSIPAÇÃO do calor para o ambiente.

Os traços pontilhados verticais delimitam a Zona de Homeotermia ou Homeostasia, em que o animal é capaz de manter a sua temperatura interna relativamente estável, porém com gasto de energia para realizar a TERMÓLISE e a TERMOGÊNESE.

Abaixo do pontilhado a esquerda do TCI, o organismo não consegue aporte de energia térmica suficiente para compensar as perdas e a temperatura corporal começa a declinar rapidamente.

De maneira análoga, acima do pontilhado a direita do TCS não consegue obter resfriamento necessário para a manutenção do equilíbrio homeotérmico e a temperatura corporal aumenta cada vez mais.

2.2 Índices de Conforto Térmico na Produção Animal

As condições do ambiente influenciam diretamente, tanto na produção animal, como no desempenho de tarefas e de atividades do ser humano. Os índices de conforto térmico combinam dois ou mais elementos climáticos, buscando determinar a sua influência sobre o conforto, e caracterizar distintos ambientes térmicos. A escolha desses índices está relacionada diretamente com a atividade desenvolvida pelo animal, com as condições do ambiente em questão e com a importância desses aspectos.

Na literatura são encontrados diferentes índices de conforto usados na avaliação do desempenho térmico dos animais. Os mais frequentes são: Índice de Temperatura e Umidade (ITU), Índice de Temperatura Umidade e Velocidade (ITUV) e Índice de Globo Negro e Umidade (IGNU).

2.2.1 Índice de Temperatura e Umidade (ITU)

É o índice de conforto térmico mais utilizado para avaliação de alojamentos dos animais que produzem carne, leite e ovos. Leva em consideração a temperatura do termômetro de bulbo seco e mais uma medida de umidade (temperatura de bulbo úmido, temperatura de ponto de orvalho ou umidade relativa) para estabelecer o relacionamento com o desempenho dos animais (KELLY e BOND, 1971). É expresso em termos adimensionais pelas equações abaixo.

$$\text{ITU} = 0,72 (t_{bs} + t_{bu}) + 40,6 \quad \text{Eq. 1}$$

$$\text{ITU} = t_{bs} - 0,55 (1 - UR) (t_{bu} - 28) \quad \text{Eq. 2}$$

$$\text{ITU} = t_{bs} + 0,36t_{bu} + 41,2 \quad \text{Eq. 3}$$

Onde:

t_{bs} = temperatura de bulbo seco ($^{\circ}\text{C}$);

t_{bu} = temperatura de bulbo úmido ($^{\circ}\text{C}$);

t_{po} = temperatura de ponto de orvalho ($^{\circ}\text{C}$);

UR = umidade relativa.

Para aviários de postura, Brown-Brandl *et al.* (1997) propuseram uma expressão adaptada:

$$\text{ITU} = 0,6 t_{bs} + 0,4 t_{bu} \quad \text{Eq. 4}$$

que foi empregada por Vitorasso (2009) para avaliar aviários com diferentes tipos de coberturas (telhas cerâmicas ou fibrocimento; com ou sem lanternim; diferentes tamanho de beiral). O autor registrou que, no avário com telha cerâmica e sem lanternim, o ITU apresentou valores mais elevados do que no ambiente com telhas de fibrocimento, porém com lanternim.

Considerando os limites de termo neutralidade relatados por Tinoco (1998), de 28°C para a temperatura do ar e 70% para a umidade, tem-se que o limite de conforto térmico para poedeiras, em termos de ITU, é de 26,2.

Nääs *et al.* (2002) empregaram o ITU para estudar o desempenho de vacas leiteiras alojadas em sistema de *freestall*, comparando dois lotes de animais (mais produtivas: $> 29 \text{ kg/dia}$ e menos produtivas: $< 14 \text{ kg/dia}$). No estudo, foi verificada

correlação relativamente baixa entre declínio na produção de leite e ITU, para os dois grupos de vacas estudados. Embora não tenha havido uma variação alta dos valores de ITU, a variação do declínio da produção se manteve crescente para as vacas menos produtivas, enquanto, para as mais produtivas, houve uma tendência de queda do declínio na produção de leite, com a variação do ITU. O declínio da produção para as vacas menos produtivas ocorreu com ITU próximo a 81 e, para as vacas mais produtivas, em torno de 83.

Sampaio *et al.* (2004) empregaram o ITU para avaliar o ambiente térmico em instalação para crescimento e terminação de suínos nas condições tropicais, em áreas sombreadas e não sombreadas. Os resultados foram: nas áreas sombreadas, ITU igual a 74,8 no verão e 62,1 no inverno; em áreas não sombreadas, 74,8 no verão e 61,9 no inverno. Os dados indicam que esse índice não representa a realidade das condições climáticas em regiões com alto índice de radiação, confirmando os resultados encontrados por Buffington *et al.* (1981).

2.2.2 Índice de Temperatura Umidade e Velocidade (ITUV)

As aves são animais classificados como homeotermos, pois apresentam a capacidade de manter a temperatura interna constante. Além disso, podem ser consideradas como um sistema termodinâmico aberto, por estarem em troca constante de energia com o ambiente (BAÊTA e SOUZA, 1997). Entretanto, esse mecanismo possui maior eficiência quando a temperatura do ambiente encontra-se dentro de certos limites. As aves não se ajustam, perfeitamente, em extremos de temperatura, podendo, inclusive, ter a vida ameaçada.

É importante que esses animais sejam alojados em ambientes, onde seja possível o balanço térmico (RUTZ, 1994).

Uma temperatura ambiental eficaz é o resultado da integração de diversos fatores, incluindo a temperatura de bulbo seco, umidade, velocidade do ar, radiação solar e precipitação. Para a produção de confinamento, a troca radiante e precipitação, geralmente, são insignificantes, considerando que a velocidade do ar e a umidade podem ser fatores significativos. Valores altos de umidade podem agravar o efeito adverso da temperatura elevada (SIEGEI, 1968; STEINBACH 1971) devido ao fato de que cada vez mais os animais necessitam da perda de calor latente devido ao

aumento da temperatura. A velocidade do ar desempenha um papel importante no alívio de calor para certos programas de habitação, como a ventilação de túnel.

A Equação 5 foi proposta (TAO, XIN; 2003), para expressar essa relação, resultando em um índice adimensional.

$$ITUV = (0,85 T_{bs} + 0,15 T_{bu}) V^{-0,058} \quad \text{Eq. 5}$$

Onde:

T_{bs} – temperatura de bulbo seco ($^{\circ}\text{C}$);

T_{bu} – temperatura de bulbo úmido ($^{\circ}\text{C}$);

V – velocidade do ar (ms^{-1});

2.2.3 Índice de Globo Negro e Umidade (IGNU)

Desenvolvido por Buffington *et al.* (1981), tem como fundamento a consideração da radiação solar direta e difusa, como fonte de calor e causadora de estresse. É calculado a partir da Eq. 3 do ITU, em que a temperatura de bulbo seco é substituída pela temperatura de globo negro.

$$IGNU = 0,6 T_{gn} + 0,36 T_{po} + 41,5 \quad \text{Eq. 6}$$

Onde:

T_{gn} – temperatura de globo negro ($^{\circ}\text{C}$);

T_{po} – temperatura de ponto de orvalho ($^{\circ}\text{C}$);

Para estudar diferentes tipos de telhas em instalações zootécnicas (telha reciclada à base de embalagens longa vida; telha cerâmica; telha cerâmica pintada de branco; telha de fibrocimento), Fiorelli *et al.* (2009) empregaram o IGNU, tendo em vista a importância da radiação solar nessa análise. O valor máximo do IGNU, no decorrer do dia, foi atingido em torno das 14:00h, para todas as coberturas avaliadas, constatando-se a relação entre a máxima radiação solar com o índice aplicado.

Entre os valores máximos, o protótipo com telhas cerâmicas apresentou IGNU mais baixo (82,8), enquanto o de telhas de fibrocimento, o mais elevado: 83,7.

Sampaio *et al.* (2004) avaliaram o ambiente térmico em instalação para crescimento e terminação de suínos nas condições tropicais em áreas sombreadas e não sombreadas. No verão, obtiveram médias de IGNU entre 68,9 e 74,8. No inverno,

devido ao baixo índice de radiação, os valores de IGNU tiveram uma amplitude bem inferior ao ocorrido no verão, obtendo valores médios de IGNU entre 55,3 a 61,2.

Santos *et al.* (2005) avaliaram o ambiente térmico em modelos reduzidos de galpões avícolas, com diferentes tipos de cobertura, empregando o IGNU, obtendo valores médios de 79,4 para a cobertura com telhas cerâmicas e sem lanternim, e de 80,6 para a cobertura com telhas de alumínio, também sem lanternim.

2.3 Engenharia de *Software*

O processo de construção de um sistema é uma atividade contemplada pela Engenharia de *Software* (ES). Como tal, precisa seguir um conjunto de métodos e técnicas para a correta construção do produto, no caso, um programa computacional. O objetivo deste subitem é descrever essa metodologia, destacando os seus principais aspectos, proporcionando um conhecimento que possa ser efetivamente utilizado para a melhoria do processo de criação de programas computacionais, com benefícios diretos para os desenvolvedores e seus clientes (usuários e colaboradores).

2.3.1 Paradigmas da Engenharia de *Software*

Bauer (1972) define a ES como: "O estabelecimento e uso de sólidos princípios de engenharia para que se possa obter um *software* economicamente viável, que seja confiável e funcione eficientemente em máquinas reais". Para Pressman (1995) ainda que várias definições tenham sido dadas à ES, todas reforçam a exigência da disciplina no desenvolvimento de *software*. Abrangendo um conjunto de três elementos fundamentais: métodos, ferramentas e procedimentos. Os métodos detalham o como fazer para se construir o *software*, as ferramentas proporcionam apoio automatizado aos métodos e os procedimentos constituem o elo entre os métodos e suas ferramentas, possibilitando um processo de desenvolvimento claro e eficiente, visando garantir ao desenvolvedor e seus clientes a produção de um *software* de qualidade.

Nos últimos 20 anos, o *hardware* deixou de ser o item mais caro na implementação de um sistema, enquanto o custo relacionado ao *software* cresceu e tornou-se o principal item no orçamento da computação. Isso se deve à crescente

complexidade dos problemas a serem resolvidos pelos *softwares*. Sistemas como os de cálculos científicos e sistemas de *enterprise resource planning* (ERP) chegam a possuir milhões de linhas de código e envolvem uma equipe complexa para o seu desenvolvimento (DEGOULET e FIESCHI, 1997). Aliado a isso, alguns problemas inerentes ao processo de desenvolvimento do *software* começaram a surgir (PRESSMAN, 1995): as estimativas de prazo e custo são imprecisas, a produtividade das pessoas da área de *software* não tem acompanhado a demanda por seus serviços e a qualidade de *software* às vezes é menor que a adequada, ocorrendo em muitos casos a insatisfação do usuário. A chave para se vencer esses problemas e dificuldades acima relatados é a larga utilização de uma abordagem de engenharia no desenvolvimento de *software*, aliada a uma contínua melhoria das técnicas e ferramentas, no intuito de melhorar a produtividade da equipe (PRESSMAN, 1995; DEGOULET e FIESCHI, 1997). Dessa forma, podem-se destacar duas tendências para justificar o uso da ES: o *software* é um item de alto custo e em progressivo aumento; tem um importante papel no bem-estar da sociedade (BOEHM, 1981). Dessa forma, a ES assume papel crítico para garantir que tarefas, dados, pessoas e tecnologia estejam apropriadamente alinhados para produzir um sistema eficaz e eficiente (MURPHY, HANKEN e WATERS, 1999).

No processo de desenvolvimento de um *software*, um conjunto de etapas deve ser definido, é denominado de Paradigmas da Engenharia de Software (PRESSMAN, 1995); também conhecido como Modelos de Ciclo de Vida de Software. Destacam-se alguns paradigmas (DAVIS, 1997): o ciclo de vida clássico (Figura 6), o modelo incremental, o evolucionário, o concorrente, a prototipação e o modelo espiral. Deve ser lembrado ainda que os paradigmas sejam combinados (Figura 7), obtendo-se um melhor resultado.

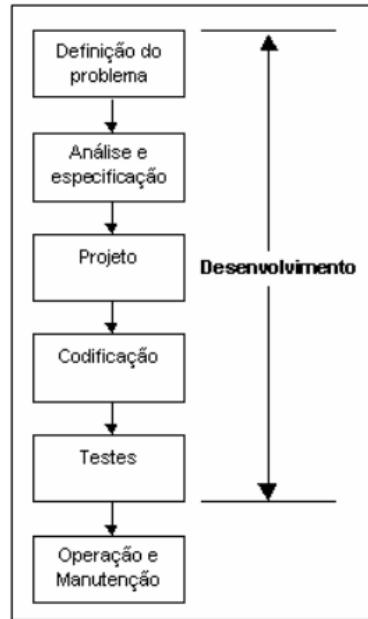


Figura 6 - Ciclo de vida clássico do software

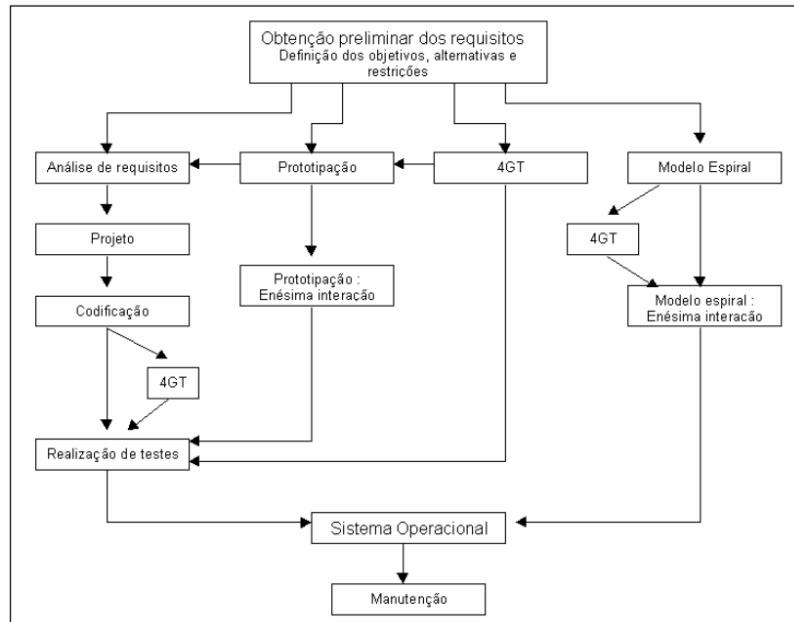


Figura 7 - Combinando Paradigmas da ES

Independentemente do paradigma a ser utilizado, três fases genéricas dividem o processo de desenvolvimento (PRESSMAN, 1995):

- Definição: essa fase focaliza o quê o *software* deverá fazer (análise do sistema, planejamento do projeto de *software* e análise de requisitos).
 - Desenvolvimento: focaliza-se no como as tarefas serão executadas pelo *software* (projeto de *software*, codificação e realização de testes).

- Manutenção: concentra-se nas mudanças pelas quais o *software* passará enquanto estiver em uso (correção, adaptação e melhoramento funcional).

A Prototipação traz bons resultados, principalmente quando o cliente não tem exatidão na declaração do problema. Como exemplo, a construção de protótipos em projeto de prontuário eletrônico de pacientes é extensamente utilizada, pois são sistemas complexos e necessitam de um alto nível de colaboração do usuário no processo de desenvolvimento (ADELHARD *et al.*, 1995).

O modelo espiral é baseado no princípio do desenvolvimento incremental, em que novas funções são adicionadas a cada ciclo de desenvolvimento. Análise, especificação, projeto, implementação e homologação são repetidas a cada ciclo, gerando uma nova versão do *software* e permitindo um *feedback* imediato do usuário (DEGOULET e FIESCHI, 1997).

Existem também as Técnicas de Quarta Geração (4GL) que utilizam poderosas ferramentas para o desenvolvimento do *software*, permitindo um nível de especificação mais elevado, próximo à linguagem natural, sendo capazes, a partir dessas definições, de gerar o código fonte do sistema (PRESSMAN, 1995).

Existem no mercado diversas metodologias de ES que criaram novos paradigmas, combinando e aproveitando os melhores conceitos das outras metodologias. Nesse ponto deve-se destacar a metodologia *Rational Unified Process* (RUP) da Rational Inc. como sendo uma das principais metodologias utilizadas atualmente no mundo, principalmente em conjunto com a ferramenta *Computer - Aided Software Engineering* (CASE), Rational Rose (JACOBSON, BOOCH e RUMBAUGH, 1999).

2.3.2 Gerenciamento de Projetos

O gerenciamento de projetos deve abranger todo o desenvolvimento, sendo praticado em cada etapa do processo. Uma das primeiras atividades de gerenciamento é o chamado Estudo de Viabilidade. Sua proposta é justificar a necessidade para o desenvolvimento do sistema, tanto do ponto de vista técnico e organizacional como financeiro (custos), através do estudo de índices como retorno sobre Investimento (DAVIS, 1998). No gerenciamento, deve-se destacar o uso das métricas de *software* que são usadas para medir a qualidade dos softwares e controlar

a produtividade dos projetos (RAMAMOORTHY *et al.*, 1984). As métricas podem ser categorizadas como diretas ou indiretas, da produtividade e da qualidade, orientadas ao tamanho, ou à função e às pessoas (VON MAYRHAUSER, 1990; ABRAN e ROBILLARD, 1996).

O planejamento, como atividade de gerenciamento, deve ocorrer baseado em estimativas seguras que possuem a experiência passada de outros sistemas como único guia. Surge, então, a necessidade de efetivamente usar-se métricas de *software*, para se construir uma grande base de informações, para serem utilizadas em projetos futuros. Diversos modelos empíricos de estimativa foram desenvolvidos, destacando-se o *constructive cost model* (COCOMO) (Boehm, 1981) e o modelo de estimativa de Putnam (MURPHY, HANKEN e WATERS, 1999). Há também, modelos gráficos para o controle do cronograma do projeto, tais como Gantt e PERT (MURPHY, HANKEN e WATERS, 1999). Entretanto, o uso de métricas é uma atividade complexa e não se têm os registros da experiência passada de softwares similares, tornando essa atividade ainda mais difícil e incerta.

2.3.3 Requisitos e Análise

O primeiro passo na construção de um sistema deve ser o entendimento de "o quê" será desenvolvido, através do levantamento dos requisitos e sua análise. Os requisitos se referem às necessidades dos usuários, do sistema, de custos e prazos. A especificação dos requisitos é de suma importância, pois a maior parte dos erros encontrados durante os testes e a operação dos sistemas são derivados de pouco entendimento ou má interpretação dos requisitos (RAMAMOORTHY *et al.*, 1984; DEGOULET e FIESCHI, 1997). Com isso, é de fundamental importância a compreensão total dos requisitos para que se obtenha sucesso no desenvolvimento de um *software*, gerando benefícios como: a aceitação de todos os envolvidos (usuários, desenvolvedores, etc.), servirá de base para estimativas (custos, prazos, equipe, etc.), melhora da usabilidade, melhoria da manutenibilidade e de outros atributos de qualidade do sistema, cumprimento das metas com os recursos previstos, dentre outros (DORFMAN, 1997). A análise de requisitos visa, também, garantir uma estrutura de dados adequada, para que futuras aplicações, tais como ensaios clínicos, por exemplo, possam ser implementados e contar com todas as informações necessárias (DOLIN, 1994).

Após os requisitos, segue-se a análise do problema a ser informatizado. Existem diversas técnicas para análise e modelagem de sistemas, tais como: análise estruturada, análise orientada a objeto (OOA - *Object-Oriented Analysis*) e modelagem de dados. Atualmente, destaca-se a OOA que introduziu uma série de novos conceitos. A OOA traz vários benefícios, como: funcionalidades complexas podem ser desenvolvidas com uma codificação menor e melhor; um rápido desenvolvimento é alcançado comparado a outros métodos e as aplicações são mantidas mais facilmente (Davis, 1998).

Com a evolução dos processos, sentiu-se a necessidade de se ter uma linguagem unificada que se tornasse poderosa o suficiente para modelar qualquer tipo de aplicação. Dessa necessidade surgiu a *Unified Modeling Language* (UML), uma linguagem padrão para especificar, visualizar, documentar e construir artefatos de um sistema. Grandes grupos tais como o *Object Management Group* (OMG) aprovam e estão envolvidos no processo de normatização dessa linguagem (FURLAN, 1998).

2.3.4 Projeto e Implementação

Enquanto as fases de requisito e análise concentram-se no “o quê” a solução fará, o projeto descreve “como” o *software* será implementado (VON MAYRHAUSER, 1990). A fase de projeto também pode ser vista como um aprofundamento da análise caminhando em direção à implementação do sistema. É durante a fase de projeto que a estrutura geral, o estilo do sistema e a sua arquitetura, são definidos (RUMBAUGH *et al.*, 1994).

Da mesma forma da análise, existem diversos métodos para o projeto do *software*, cada qual com o seu conjunto de princípios e notações. Dentre vários podem ser citados: projeto orientado ao fluxo de dados, *Object Oriented Design* (OOD), projeto estruturado e o desenvolvimento estruturado de Jackson (Pressman, 1995). Entretanto, o OOD atualmente tem tomado a atenção dos desenvolvedores, que o têm utilizado largamente, e em conjunto com a OOA, constitui uma metodologia completa: a *Object Oriented Analysis and Design* (OOAD) (MARTIN e ODELL, 1992). Vários trabalhos citam o desenvolvimento de programas computacionais utilizando projeto orientado a objeto (DOLIN, 1994; SAKAMOTO, 1998). Com o OOAD é possível a utilização de *Design Patterns* (padrões de projetos) e *Frameworks* (modelos de arquiteturas). *Design Patterns* são estruturas que aparecem repetidamente nos

projetos orientados a objeto para resolver um determinado problema de forma flexível e adaptável dinamicamente, trazendo vantagens como o aumento da produtividade e da consistência das aplicações. Podem ser combinados para resolverem problemas mais complicados e, a cada dia, novos padrões de projeto surgem, enriquecendo o conjunto de opções para o desenvolvedor. Um *framework* pode ser considerado como uma infraestrutura de classes que oferecem o comportamento necessário para implementar aplicações dentro de um domínio específico, funcionando com um molde para as aplicações (GAMMA, 1995).

Além do projeto de dados orientado a objetos, deve-se elaborar um projeto de interface, que estabeleça o *layout* e os mecanismos de interação homem-máquina (PRESSMAN, 1995). O estudo da interface é importante no desenvolvimento de um programa computacional para se conquistar a adesão dos usuários. Assim, conceitos como usabilidade e ergonomia, bem como uma série de recomendações para a interface, tornam-se importantes para garantir o sucesso de um sistema (SITTIG, KUPERMAN e TEICH, 1996; ASH, 1997). A interface deve ser intuitiva, fácil de usar e acessível para os não especialistas, deve ser adaptada ao método de trabalho dos usuários, e ainda ser capaz de representar os diversos tipos de mídia (texto, gráficos, imagens e sons) (DEGOULET e FIESCHI, 1997).

Após o projeto, segue-se a codificação, também chamada de implementação. Essa fase é uma simples questão de tradução do projeto para um código, já que as decisões mais difíceis já foram tomadas durante a fase de projeto (RUMBAUGH *et al.*, 1994). Existem hoje, as ferramentas do tipo *Rapid Application Development* (RAD) que permitem ao usuário um rápido desenvolvimento, baseado em conceitos de reusabilidade e componentização. Java, Visual Basic, Delphi e C++ são algumas das linguagens de programação mais usadas atualmente. Além disso, tecnologias específicas para o desenvolvimento de sistemas na Web, tais como o *Active Server Pages* (ASP), têm sido muito utilizadas nos últimos anos.

2.3.5 Testes, Manutenção e Qualidade de Software

Várias estratégias de testes podem ser implementadas para assegurar que o software está em acordo com suas especificações e livre de erros. Teste de unidade, teste de integração, teste de sistema, teste de instalação e teste de aceitação são exemplos de técnicas que podem ser utilizadas (VON MAYRHAUSER, 1990). Os mais

conhecidos são: o *alpha-test*, em que o *software* é testado num ambiente controlado por alguns usuários e na presença dos desenvolvedores; e o *beta-test*, em que o *software* é testado por um conjunto maior de usuários, que se propõem a dar um *feedback* aos desenvolvedores, caso alguma irregularidade seja encontrada. Muitas ferramentas CASE oferecem suporte automatizado ao processo de teste. A estratégia para a realização dos testes em programas computacionais também deve medir a efetividade do sistema, avaliando itens como: consistência, confiabilidade e validação de conteúdo (MILHOLLAND, 1995).

A manutenção consome mais de 60% do custo no ciclo de vida de um *software*. Isso ocorre devido ao fato dos programadores serem negligentes durante as fases anteriores a implementação do *software* (RAMAMOORTHY *et al.*, 1984). Dessa forma, para uma manutenção mais tranquila e segura, deve-se utilizar extensamente a ES que garantirá um *design* adequado e escalável para futuras modificações. Durante a manutenção, são realizadas atividades corretivas, adaptativas e preventivas (PRESSMAN, 1995). A manutenção de um sistema deve ser feita de maneira criteriosa, para evitar que alterações inseridas venham a prejudicar o funcionamento do sistema e implique em erros que possam causar sérios danos, principalmente ao paciente, como por exemplo, na disponibilização de informações incorretas.

A ES é a responsável pelo controle da qualidade, fazendo com que o sistema atenda a todos os requisitos e atributos (RAMAMOORTHY *et al.*, 1984), assumindo assim papel crítico na produção dos sistemas. A *Software Quality Assurance* (SQA) é uma atividade que deve ser aplicada ao longo de todo o processo de desenvolvimento, envolvendo revisões técnicas formais, múltiplas fases de teste, controle da documentação de *software* e das mudanças, procedimentos para garantir a adequação aos padrões e mecanismos de medição e divulgação (PRESSMAN, 1995).

A avaliação de um sistema de informação é um processo que deve ser feito por toda a "vida" do *software*, devendo-se avaliar itens como: desempenho, custo efetividade, aceitação do usuário e segurança (PERREAUXT e WIEDERHOLD, 1990).

Um dos métodos mais modernos, que visam o aumento da qualidade de *software* é o modelo proposto pelo *Software Engineering Institute* (SEI) da Carnegie Mellon University conhecido como *Capability Maturity Model* (CMM), que visa a melhoria da qualidade de serviços de desenvolvimento de programas de computador. Esse modelo divide em cinco níveis a situação de amadurecimento no processo de desenvolvimento de *software* de uma organização, bem como oferece os meios de se

atingir esses estágios. Segundo o SEI, apenas 1% das empresas se encontram no nível 4 ou no 5 (SEI, 1995; VAZ, 2000).

2.3.6 Orientação a Objetos

A forma mais moderna de abordagem no desenvolvimento de sistemas é a OOAD. Rumbaugh *et al.*, (1994) definem orientação a objetos como: "uma nova maneira de pensar os problemas utilizando modelos organizados a partir de conceitos do mundo real. O componente fundamental é o objeto que combina estrutura e comportamento em uma única entidade". Dizer que um *software* é orientado a objetos significa que é organizado como uma coleção de objetos separados, incorporando tanto a estrutura como o comportamento dos dados. A Orientação a Objetos (OO) trouxe vários novos conceitos ao desenvolvimento de *software*, como: Abstração, Encapsulamento, Objeto, Classe, Atributo, Operação, Método, Mensagem, Evento, Interface, Generalização, Herança e Polimorfismo (JACOBSON *et al.*, 1996; FURLAN, 1998; FUZION, 1999).

Quando bem empregada, a OO traz diversas vantagens: reutilização, confiabilidade, modelo de sistema mais realístico, facilidade de interoperabilidade e de manutenção, aumento da qualidade, maior produtividade e unificação do paradigma (da análise a implementação) (MARTIN e ODELL, 1992; JACOBSON *et al.*, 1996; FUZION, 1999).

A OO é um paradigma que pode ser aplicado ao longo de todo o processo de construção do *software*. Dessa forma, tem-se as metodologias acima descritas, que atuam no processo de análise e projeto e, no ciclo de implementação, existem as tecnologias de *back-end* (banco de dados) e as de *front-end* (linguagens e ferramentas de programação) (FURLAN, 1998). Os Bancos de Dados têm evoluído no sentido de suportar a tecnologia OO. Inicialmente, foram lançados bancos objeto-relacionais que suportam apenas alguns dos conceitos OO, mantendo a estrutura do modelo relacional. Em seguida, surgiram os bancos de dados realmente OO, tais como o Jasmine da CA Computer Inc., que suportam os conceitos OO. Entretanto, devido a questões como a falta de habilidade em OO pelas empresas, dentre outras, esses bancos de dados tiveram pouca penetração no mercado (BELLOQUIM, 2000). Com isso, continuou o domínio no mercado dos bancos de dados relacionais e objeto-relacionais, forçando de certa forma os desenvolvedores a "quebrar" o paradigma da

OO no momento de implementar o banco de dados, tendo-se que utilizar técnicas de mapeamento objeto-relacional para acomodar os dois modelos no sistema. As primeiras linguagens de programação orientadas a objetos apareceram em meados de 1966, como o Simula e, em 1972, o *Smalltalk*. Linguagens com maior penetração no mercado, tais com Pascal e C, evoluíram e criaram versões OO, como o C++, por exemplo. Outras linguagens, já criadas dentro do conceito da OO, como o Java, por exemplo, possibilitaram uma maior difusão do uso dessa tecnologia pelo mercado. Viu-se, também, o rápido crescimento de ambientes de desenvolvimento integrados, que permitem a construção visual dos sistemas de forma bastante eficiente e com uso de componentes previamente montados. São exemplos dessas linguagens/ferramentas o Visual Basic e o Delphi.

A OO tem sido bastante utilizada, existindo várias publicações a respeito (DOLIN, 1994) do seu uso, suas vantagens, o ganho de produtividade, a melhoria na qualidade do produto, a reutilização de código, dentre outros pontos positivos (SETO, KAMIYAMA e MATSUO, 1998). É importante também destacar que novas tecnologias surgiram com a evolução da OO.

Talvez, o melhor exemplo sejam os objetos distribuídos, com tecnologias como DCOM e CORBA. A construção de componentes reutilizáveis foi outro grande benefício trazido pela OO. É possível construir softwares através do acoplamento de diversos componentes que embutem comportamentos específicos e que podem ser reutilizados em outros softwares. É possível, também, a criação de componentes para um mercado específico, como o da área de saúde. Destaca-se, no Brasil, o Consórcio de Componentes de Softwares para Sistemas de Informação em Saúde (CCS-SIS) que disponibiliza gratuitamente componentes que podem ser utilizados na construção de softwares para a área da saúde (Leão *et al.*, 1998). Além desse, há o Consórcio de Tecnologia da Informação na Área de Saúde (COTIS), que visa o desenvolvimento dos componentes do Sistema de Informação Unificado de Saúde (SIUSB), em conformidade com as diretrizes do Ministério da Saúde e as necessidades do mercado de saúde. É uma iniciativa das empresas de tecnologia que atuam nesse segmento (COTIS, 2001).

2.3.7 Unified Modeling Language (UML)

Nos últimos anos, com a crescente complexidade das metodologias da ES, surgiram as ferramentas CASE. Elas ajudam em todo o processo de desenvolvimento, desde o gerenciamento e análise e até mesmo na codificação.

Muitos foram os métodos desenvolvidos para a aplicação da orientação a objetos no processo de análise e projeto, entretanto, era necessário um caminho comum. Com o decorrer do tempo, as metodologias de Booch, a OMT (de Rumbaugh) e a OOSE (de Jacobson) evoluíram, seus autores se uniram e lançaram uma linguagem de notação unificada, chamada UML e também lançaram uma metodologia orientada a objetos chamada RUP (JACOBSON, BOOCH e RUMBAUGH, 1999), que abrange todo o processo de desenvolvimento de um sistema. Paralelamente, James Rumbaugh e Grady Booch combinaram suas metodologias: OMT e Booch, respectivamente, através da *Rational Corporation*, nos Estados Unidos, e criaram um método comum: o *Unified Method* (UM), lançado em 1995. Em seguida, deu-se a adesão de Ivar Jacobson, outro grande metodologista, contribuindo com as ideias de sua metodologia OOSE. Esses três personagens lançaram, então, a *Unified Modeling Language* (UML) versão 0.9 em 1996. A UML versão 1.1 foi submetida ao *Object Management Group* (OMG) e aprovada como padrão mundial de linguagem de notação de projetos OO.

O objetivo da UML é prover uma linguagem padrão que permita modelar um sistema e dotar o mercado mundial de orientação a objetos de uma linguagem única de modelagem, que permita a troca de modelos de forma natural entre os construtores de softwares (FUZION, 1999). Com a UML é possível (MATTIAZZI, 1998): descrever eficazmente requisitos de software, caracterizar a arquitetura (lógica e física) de um sistema, focalizar na arquitetura em vez da implementação e direcionar programadores, aumentando a produtividade e diminuindo os riscos.

Segundo Furlan (1998), "a UML é uma linguagem de modelagem, não uma metodologia".

Na construção de um software, a UML deve ser usada em conjunto com uma metodologia de ES orientada a objetos, tal como o RUP. Não se recomenda a utilização do paradigma clássico (*waterfall*) visto que a UML se adapta melhor com paradigmas incrementais e similares. Por outro lado, a UML não se restringe a diagramas, apresenta uma série de conceitos e recursos que facilitam a identificação

de objetos e classes, associando-os aos requisitos do sistema, e oferece formas de planejar e gerenciar projetos baseados nesses requisitos (MATTIAZZI, 1998). A UML apresenta os seguintes diagramas que, em conjunto, modelam todo o sistema (MATTIAZZI, 1998; FURLAN, 1998; FUZION, 1999):

- Diagrama de Classe: utilizado para representar as diversas classes de objetos do sistema, seus atributos, operações e a associação entre cada uma delas - herança, generalização, composição e agregação (Figura 8).
- Diagrama de Caso de Uso: usado para demonstrar o relacionamento entre atores e casos de uso (Figura 9).
- Diagramas de Sequência: tipo de diagrama de interação que apresenta a interação de sequência de tempo dos objetos participantes da interação (Figura 10).
- Diagrama de Colaboração: tipo de diagrama de interação que mostra uma interação dinâmica de um caso de uso e seus objetos relacionados.
- Diagrama de Estado: utilizado para demonstrar as sequências de estados que um objeto assume em sua vida, em função do seu uso no sistema.
- Diagrama de Atividade: tipo de diagrama de estado em que a maioria dos estados são ações. Descreve o fluxo interno de uma operação.
- Diagrama de Componente: usado para representar os diversos componentes dos sistemas e suas dependências.
- Diagrama de Implantação: utilizado para demonstrar elementos de configuração de processamento *runtime*.

O uso de um tipo ou outro de diagrama depende, muitas vezes, do grau de detalhamento necessário para o desenvolvimento do sistema. Os diagramas de classe, de casos de uso e de sequência são os mais utilizados. Há ainda diversos outros conceitos, como pacote, estereótipo, dentre tantos que a UML possui, fugindo ao escopo desta dissertação a explicação de cada um desses. Para um bom uso da UML, recomenda-se a utilização de ferramentas CASE, que ajudam na construção dos diagramas, dando suporte automatizado à notação. Hoje, a maioria das ferramentas CASE já suportam a UML, sendo algumas totalmente direcionadas a ela.

A UML apresenta algumas deficiências (KUIKKA, 1999; GLINZ, 2000): não mostra precisamente os limites do sistema, não oferece uma visão geral da funcionalidade do sistema, é insuficiente para a descrição do fluxo de informações na

instituição, não provê meios adequados para representar interações complexas entre use cases.

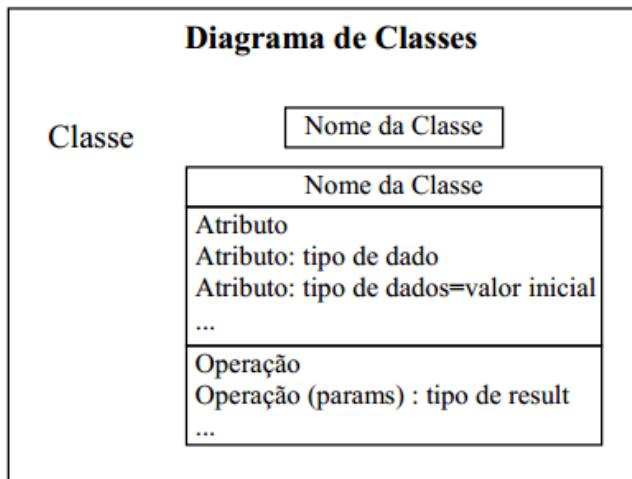


Figura 8 - Exemplo de um Diagrama de Classe da UML

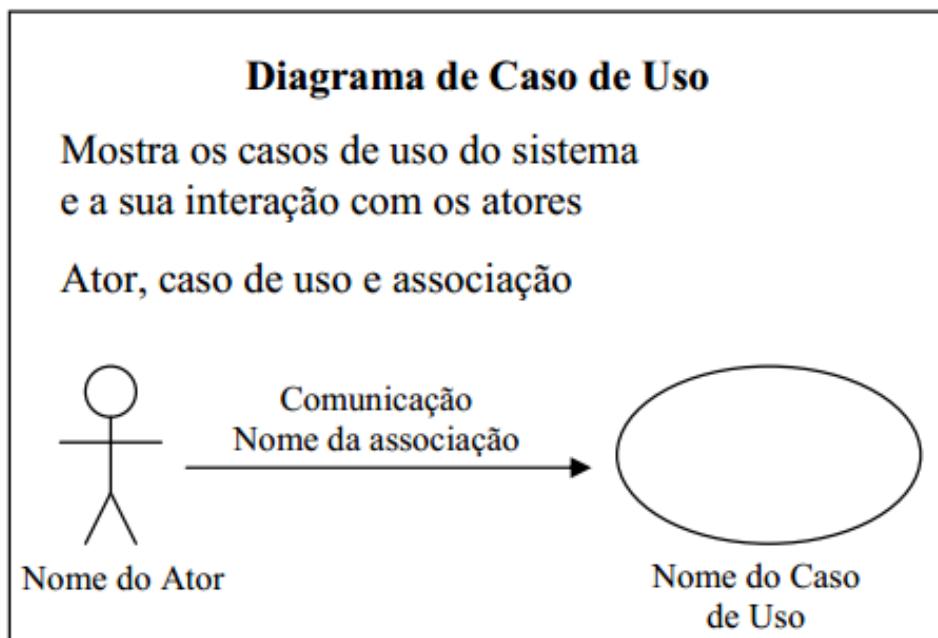


Figura 9 - Exemplo de um Diagrama de Caso de Uso da UML

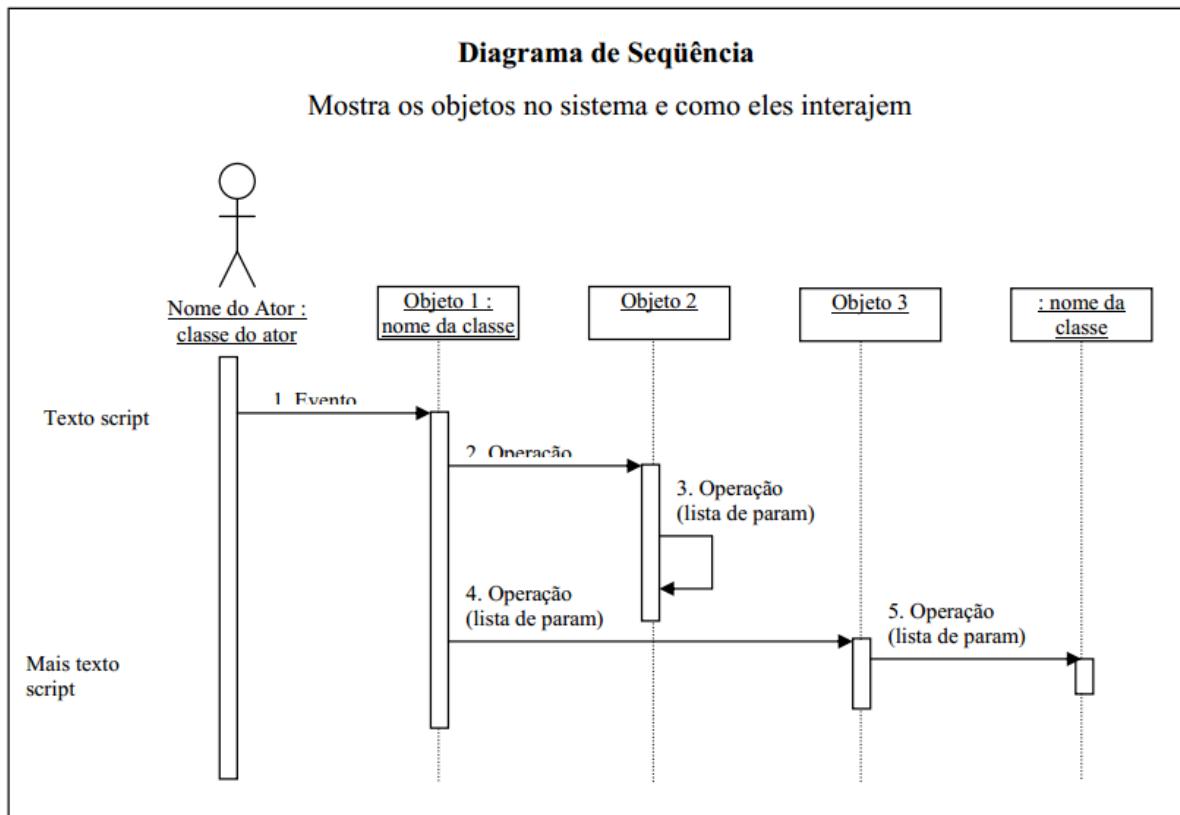


Figura 10 - Exemplo de um Diagrama de Seqüência da UML

CAPÍTULO III

3 METODOLOGIA

A metodologia deste trabalho foi desenvolvida em duas partes: (1) inicialmente foram consultados na literatura os índices relacionados ao conforto térmico na produção industrial de animais; (2) posteriormente, a partir dos algoritmos identificados foram aplicados os paradigmas da ES no desenvolvimento do programa computacional.

3.1 Organização dos Índices da Literatura

Através de pesquisas bibliográficas, foram selecionados os algoritmos que descrevem os índices de conforto térmico na produção animal para as espécies de suínos, bovinos de leite e aves, citados nas Tabelas 2 e 3.

Tabela 2 - Fonte dos algoritmos para determinação dos Índices de Conforto Térmico.

Índice	Espécie	Autor(es)
ITU	Suínos	KELLY, BOND, 1971
ITU	Bovinos	KELLY, BOND, 1971
ITU	Aves de corte	KELLY, BOND, 1971
ITUV	Aves de corte	TAO, XIN, 2003
IGNU	Suínos	BUFFINGTON <i>et al.</i> , 1981
IGNU	Bovinos	BUFFINGTON <i>et al.</i> , 1981
IGNU	Aves de corte	BUFFINGTON <i>et al.</i> , 1981

Tabela 3 - Algoritmos para determinação dos Índices de Conforto Térmico

Índice	Espécie	
ITU	$0,72 (t_{bs} + t_{bu}) + 40,6$	Eq. 1
ITUV	$(0,85 T_{bs} + 0,15 T_{bu}) V^{-0,058}$	Eq. 5
IGNU	$0,6 T_g + 0,36 T_{po} + 41,5$	Eq. 6

Os intervalos para os resultados dos índices, bem como suas respectivas denominações encontram-se na Tabela 4.

Tabela 4 - Valores limites de ITU, ITUV e IGNU

Índice		Obs.	Conforto	Alerta	Perigo	Emergência	Autor
ITU	Frangos	1	< 74	74 - 79	79 - 84	> 84	Thom (1959)
	Bovinos	2	<= 70	71 - 78	79 - 83	> 83	Hahn (1985)
	Suínos	1	61 - 65	65 - 69		69 - 73	Sales et al. (2006)
ITUV	Frangos	1	<= 24	<= 34	<= 39	> 39	Xao e Xin (2003)
	Frangos	1	76			> 76	Teixeira (1983)
IGNU	Bovinos	2	< 74	74 - 78	79 - 84	> 84	Baêta (1985)
	Suínos	1	69,6			82,6	Ferreira (2001)

(1) Animal próximo ao abate

(2) Animal em produção

Um fator de grande importância na escolha dos algoritmos foi a presença de variáveis possíveis de serem coletadas diretamente pelo produtor sem o auxílio de equipamentos sofisticados.

3.2 Fluxo dos Processos para Definição do Programa Computacional

Os procedimentos adotados para a construção do programa computacional proposto são apresentados na Figura 11.

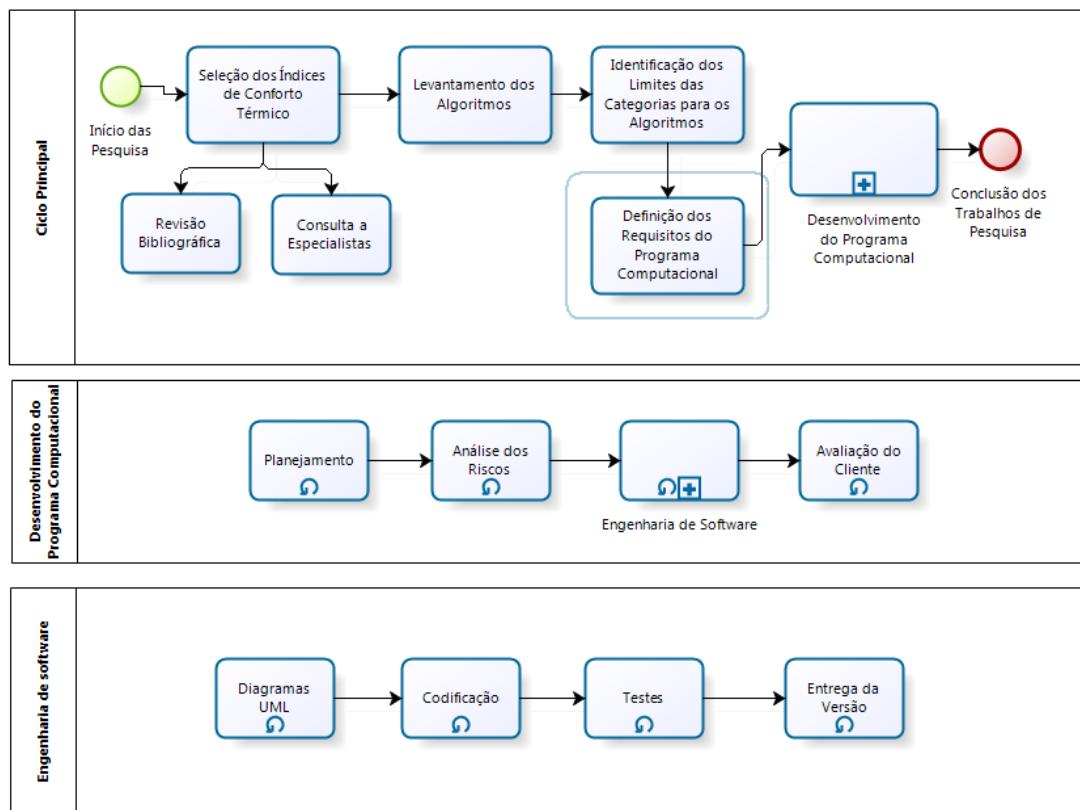


Figura 11- Fluxograma dos procedimentos para a execução da presente pesquisa

3.3 Diagramas UML

Devido ao nível de complexidade do programa computacional não ser muito elevado, optou-se pela utilização de apenas três dos diagramas disponibilizados pela UML. Os diagramas selecionados foram os de caso de uso, sequência e classes.

3.3.1 Caso de Uso

Esse diagrama é utilizado para demonstrar o relacionamento entre atores e casos de uso (Figura 12).

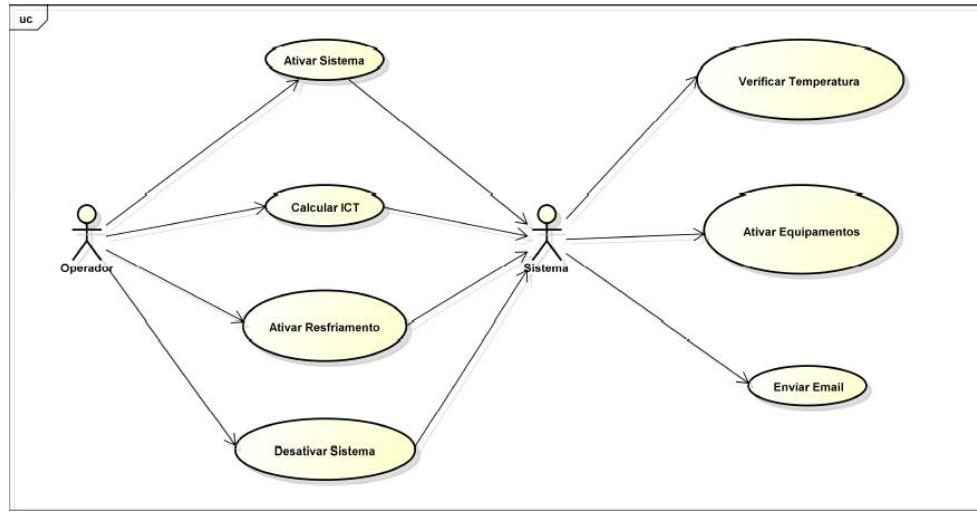


Figura 12- Exemplo de Diagrama de Caso de Uso utilizado no programa computacional

3.3.2 Sequência

Apresenta a interação de sequência de tempo dos objetos que participam na interação (Figura 13).

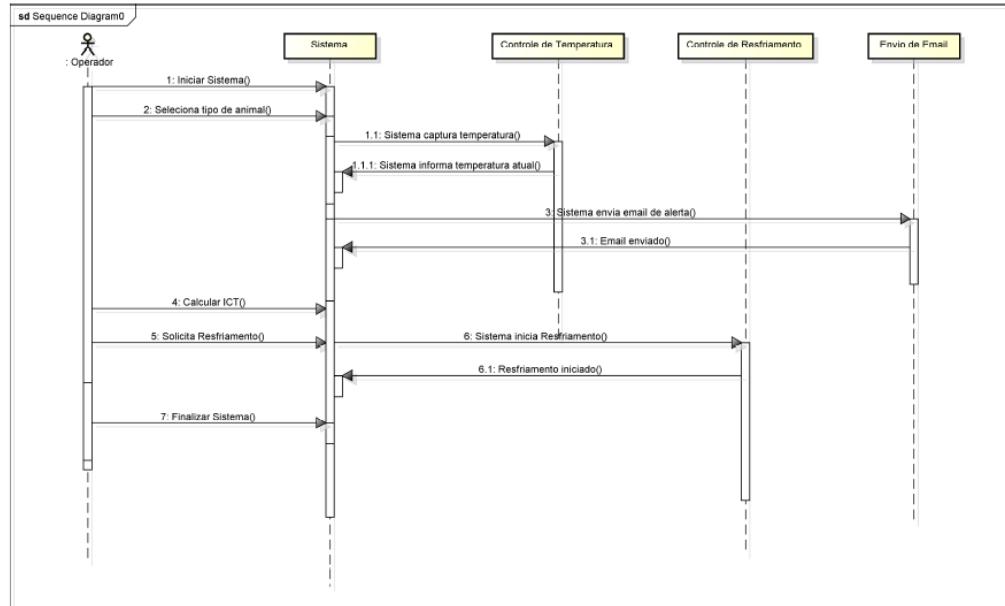


Figura 13- Exemplo de Diagrama de Sequência utilizado no programa computacional

3.3.3 Classes

Representar as diversas classes de objetos do sistema, seus atributos e operações, bem como a associação entre cada uma delas (Figura 14).

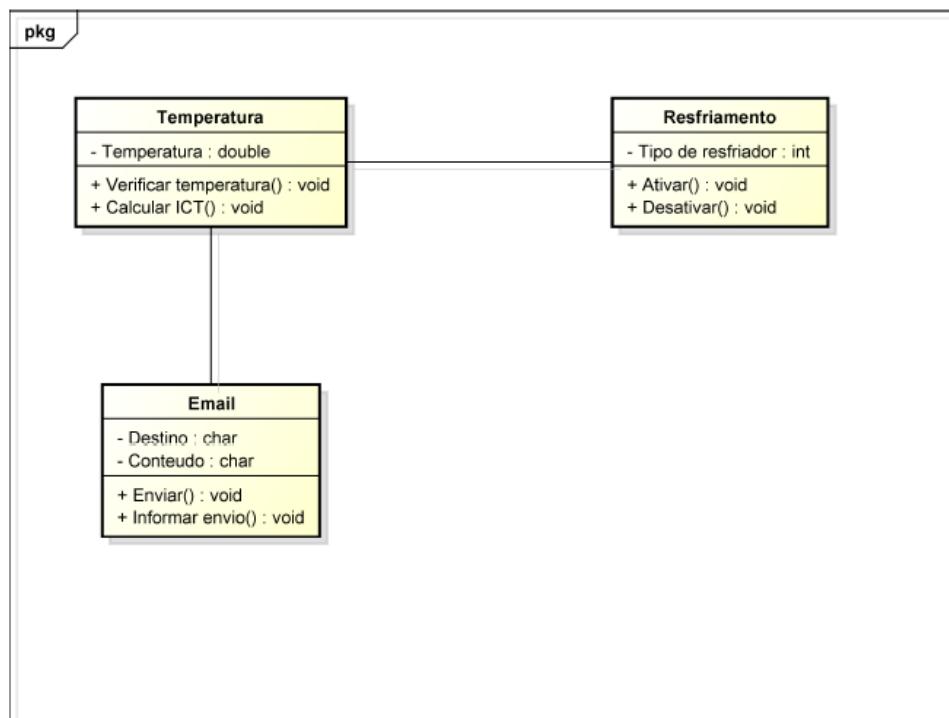


Figura 14 - Exemplo de Diagrama de Classes utilizado no programa computacional

3.4 Desenvolvimento do Programa Computacional

Uma das premissas básicas utilizadas neste trabalho foi a de que o programa computacional deveria ser de fácil operação para maximizar o seu uso pelos produtores. Para isso foram selecionados índices com indicadores de fácil obtenção. Realizou-se a revisão bibliográfica em busca de índices que contemplassem esses requisitos e fossem significativos para possibilitar controles e, possivelmente, aumentos de produtividade. A plataforma de desenvolvimento foi o Visual Studio 2012, sendo a linguagem de programação o C#. O paradigma de desenvolvimento adotado foi a orientação a objetos, e foram empregados os conceitos de ES em cada uma das fases do ciclo de desenvolvimento e testes.

Os indicadores selecionados foram ITU, ITUV e IGNU.

3.4.1 Interface Gráfica com o Usuário

Para a elaboração da interface (Figura 12), levou-se em conta a facilidade de operação pelo usuário, sendo feito um estudo para que a disposição, tamanho e cores dos controles atingissem esse objetivo.

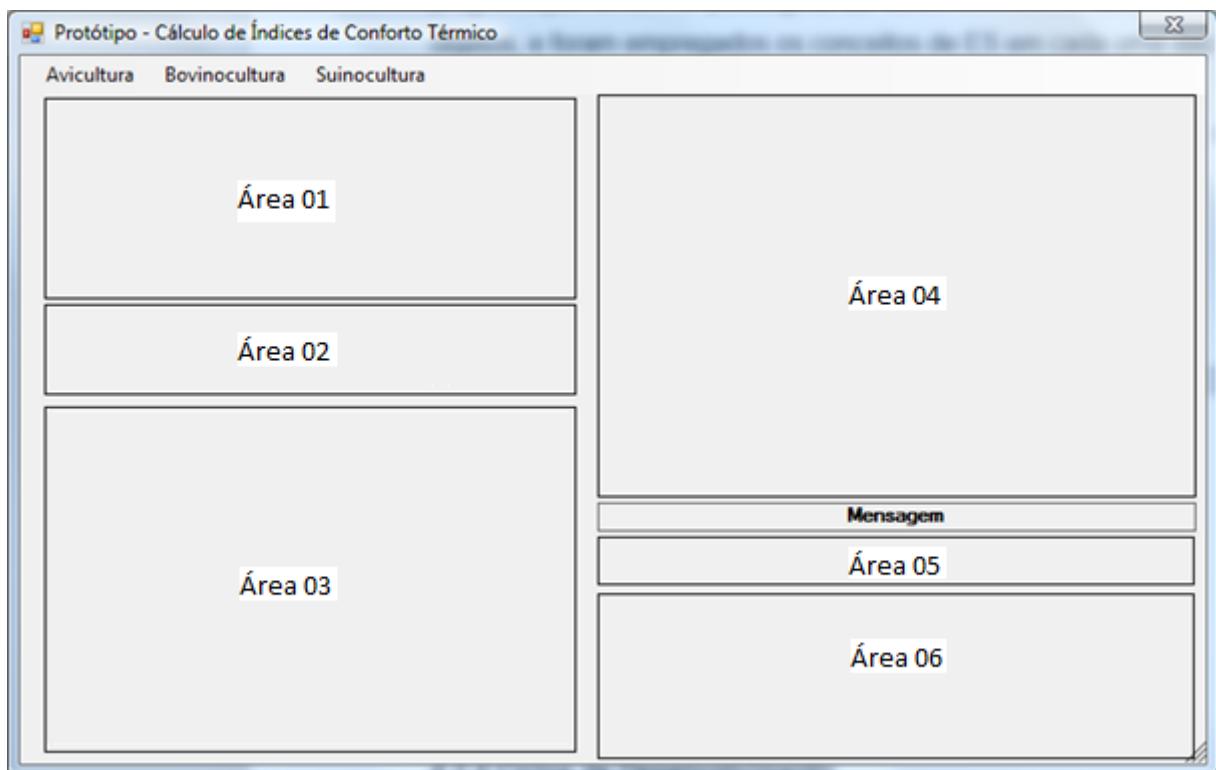


Figura 15 - Interface Gráfica com o Usuário

Será feita uma descrição de cada uma das áreas elucidando os seus conceitos e controles, bem como os objetivos aos quais se pretendia serem alcançados quando da sua elaboração.

- ✓ Área 01 – Entrada dados pelo usuário: Os valores podem ser digitados diretamente nos campos adequados ou preenchidos através dos controles auxiliares clicando-se nas setas indicativas.
- ✓ Área 02 – Configuração das opções gerais do programa: É possível ativar ou desativar as opções de coletar dados através de sensores remotos, enviar e-mails de aviso com os dados sobre os índices, tocar sons de alerta ou controlar o funcionamento remoto de equipamentos como nebulizadores e ventiladores.

- ✓ Área 03 – Exibição de imagens. São exibidas as imagens das espécies de animais para as quais os índices estão sendo calculados (aves de corte, bovinos e suínos). Também são mostradas se o programa ativou os equipamentos remotos e a intensidade do seu uso.
- ✓ Área 04 – Gráficos: O gráfico superior exibe os valores dos últimos 20 índices calculados. O gráfico inferior é preenchido com as respectivas temperaturas ou velocidade do vento, de acordo com o índice (ITU, ITUV ou IGNU).
- ✓ Mensagem: Exibe uma mensagem, com as cores variando conforme o intervalo em que o valor do índice se encontra. Os valores e cores possíveis são: normal, cor verde; alerta, cor amarela; perigo, cor vermelha e emergência, cor preta.
- ✓ Área 05 – Indicativos coloridos: Simula um semáforo, e as cores acompanham o padrão contido na Área Mensagem explicada no tópico anterior.
- ✓ Área 06 Status e Orientações – é o local onde o programa interage com o usuário através de mensagens, e podem conter informações sobre o status do programa, intervenções feitas pelo programa (no caso de sensores e equipamentos remotos) ou solicitadas ao usuário. São apresentadas sugestões para a mitigação dos riscos envolvidos quando os índices não estão nos intervalos ótimos.

Como exemplos de telas do programa computacional em operação, há as Figuras 16, 17 e 18, mostrando um exemplo para cada uma das espécies e índices disponíveis.

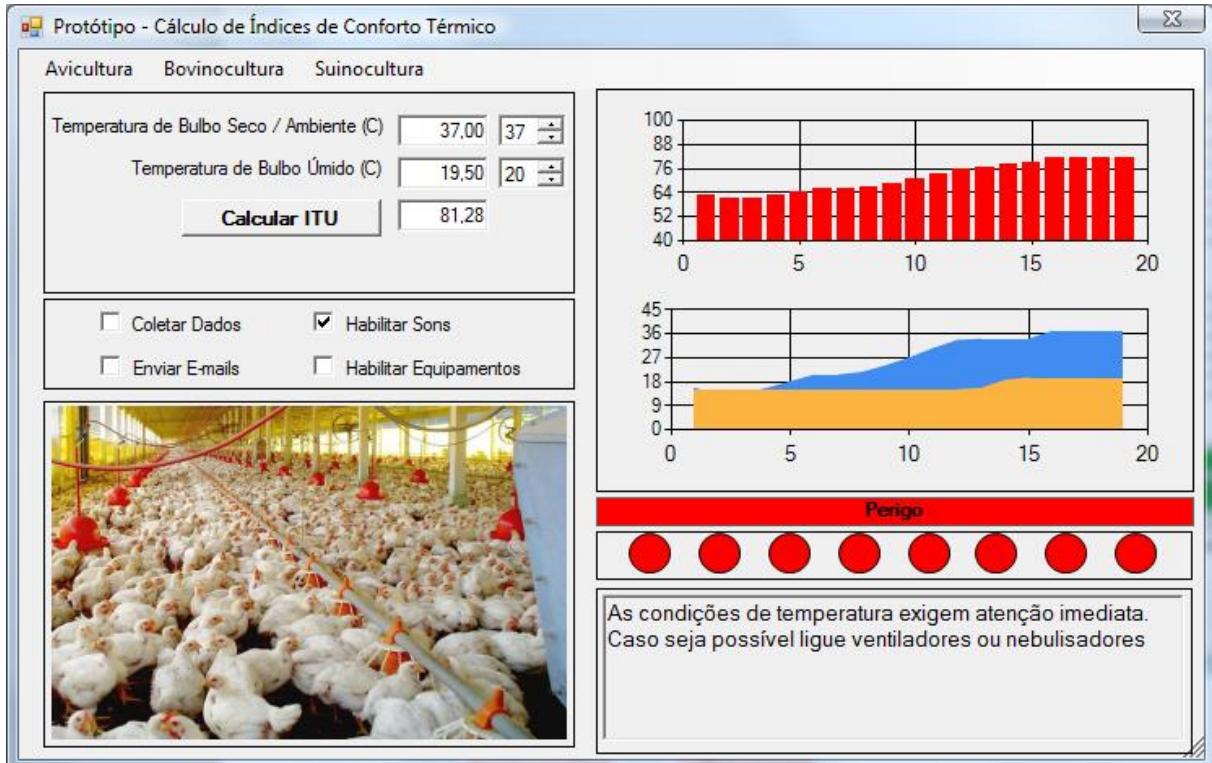


Figura 16 - Tela exemplo do cálculo do ITU para aves de corte

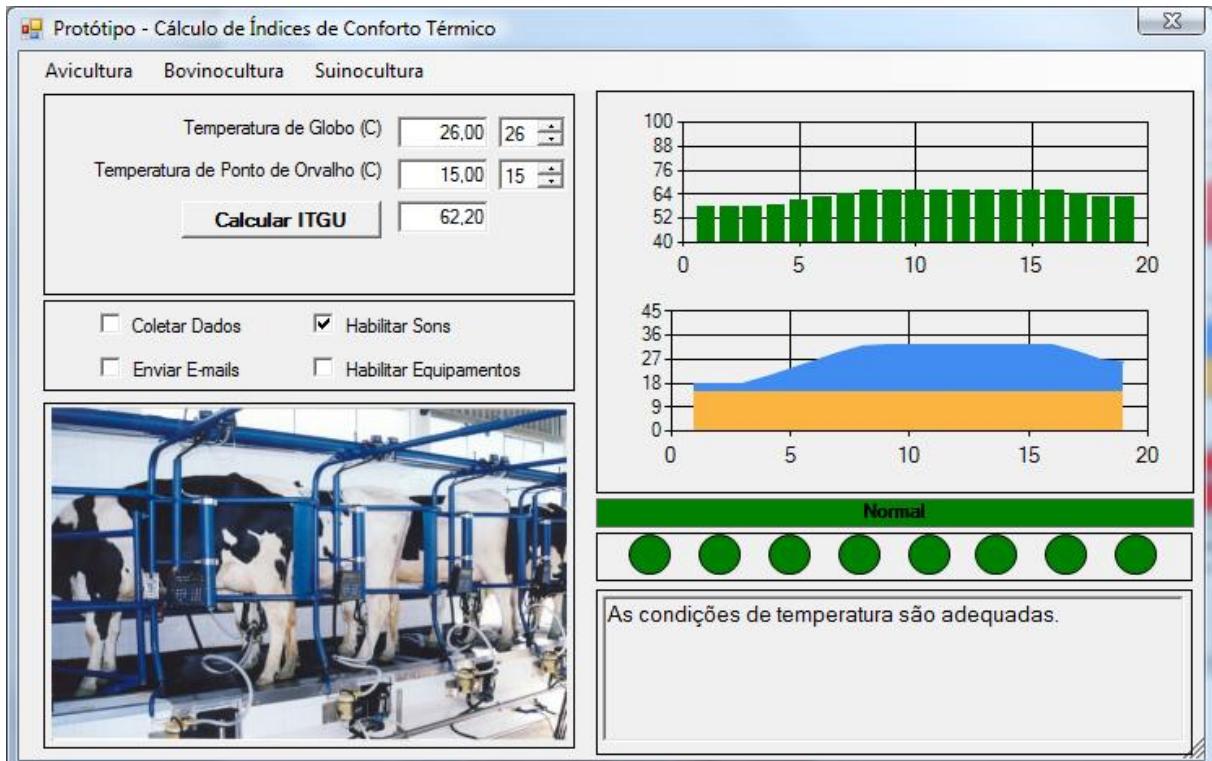


Figura 17 - Tela exemplo do cálculo do ITGU para bovinos

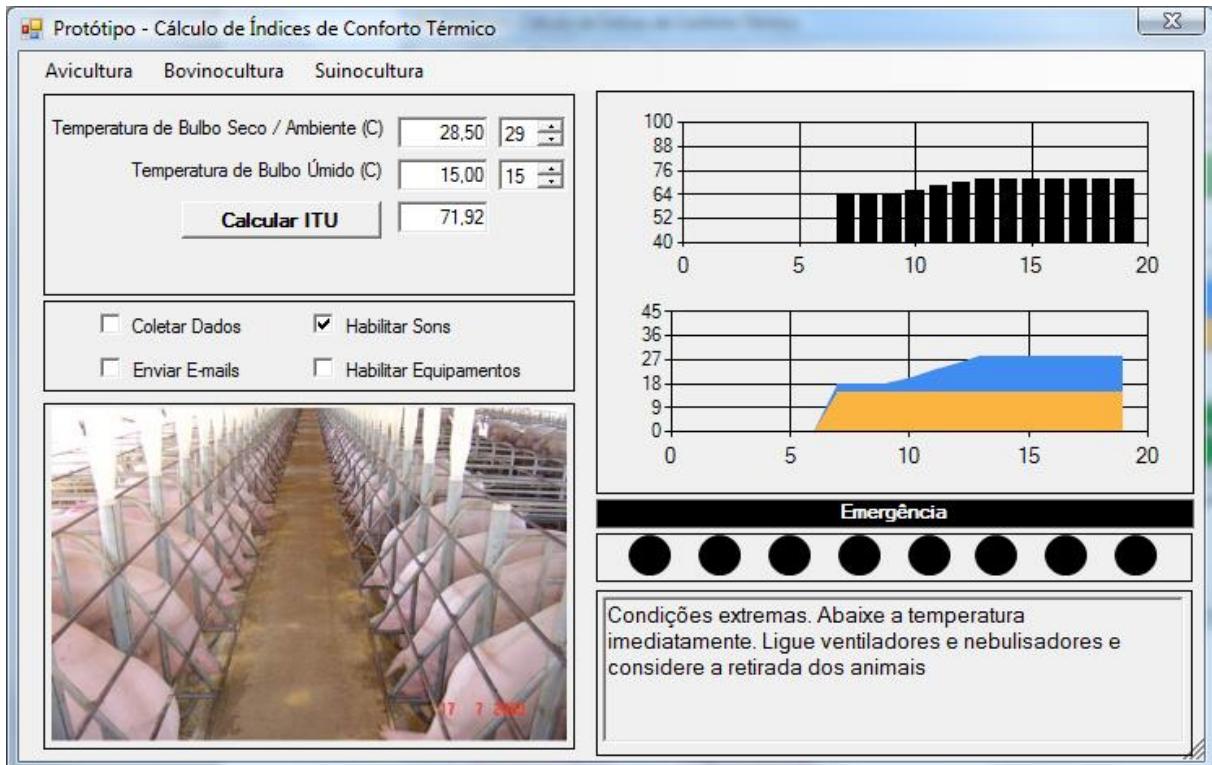


Figura 18 - Tela exemplo do cálculo do ITU para suínos

3.4.2 Codificação dos Algoritmos

As equações de cada um dos índices de conforto térmicos foram codificadas em linguagem C#. Fatores como performance, uso de memória, problemas de conversão de dados provenientes da digitação, precisão numérica, possibilidade de customização da distribuição dos resultados nos intervalos para períodos diferentes (por exemplo, nas primeiras semanas ou na fase de abate), e os intervalos específicos para cada uma das espécies de animais envolvidas neste trabalho foram todos contemplados.

Executou-se análise de performance através das ferramentas nativas no ambiente de desenvolvimento e foram feitas mudanças a partir dos resultados obtidos, até se alcançar valores adequados e a estabilidade do *software*.

3.4.3 Interface com Equipamentos de Coletas de Dados

Os fabricantes dos sensores fornecem *drivers* juntamente com os equipamentos de coleta de dados. Com isso, a codificação do programa computacional para a coleta de dados consiste na chamada a funções fornecidas

através desses *drivers*. O resultado é fornecido como uma sequência de caracteres que podem facilmente ser utilizados como entrada de dados.

3.4.4 Envio de e-mails

Outra facilidade do programa computacional é a capacidade de enviar e-mails para um usuário ou uma lista de usuários cadastrados, contendo o status atual do sistema, o valor e o nome do índice que está sendo monitorado e mensagens com orientações sobre ações a serem tomadas.

3.4.5 MultiThread

Sutter (2009) discorre sobre os conceitos de programação para a maximização do uso do processador. Em ciência da computação, um *thread* de execução é a menor sequência de instruções programadas, que podem ser gerenciadas de forma independente por um sistema operacional. Em um único processador, *multithreading* ocorre, geralmente, por multiplexação da divisão de tempo (como em multitarefa); o processador alterna entre diferentes *threads*. Essa mudança de contexto acontece com frequência suficiente para que o usuário perceba as tarefas como ocorrendo simultaneamente.

O resultado perceptível do ponto de vista do usuário é que processos que consomem muitos ciclos do processador para serem executados, não retardem a interação com a interface. No caso do programa computacional objeto deste trabalho, o envio de e-mails e a leitura dos dados a partir de sensores remotos causariam esse efeito caso fossem executados de maneira síncrona.

Para evitar esse problema foram implementados *multithreads* nesses dois processos, propiciando os ganhos já descritos, tendo como contrapartida o aumento da complexidade da codificação.

3.4.6 Ciclos de Desenvolvimento

Com base em entrevista com especialistas, os resultados do programa computacional foram validados contra a experiência dos pesquisadores e também em relação a bibliografia.

De acordo com o *feedback* dos especialistas, foram implementadas as modificações sugeridas, e liberadas novas versões do programa computacional. Ainda não se têm dados para prever com qual frequência esse ciclo será possível de ser completado, mas se estima, com base em dados de pesquisas na área de *software* que o período ideal se dê em torno de três meses, prazo em que os usuários sentem que suas necessidades estão sendo atendidas, aumentando a fidelização no uso do programa computacional.

Em um primeiro momento, o programa computacional apenas calculava os índices, a partir de *feedbacks* dos especialista, implementou-se uma espécie de semáforo, que sinaliza para o usuário em qual região o valor obtido se encontra (Conforto, Alerta, Perigo e Emergência).

Em outro ciclo, uma orientação básica referente a como mitigar os riscos envolvidos conforme a faixa em que o índice apurado se enquadrava foi desenvolvida.

No estágio atual, o programa computacional permite a digitação dos dados pelo usuário ou a coleta através de sensores remotos. Cada vez que os índices de conforto térmico entram em um dos intervalos previamente identificados, existe a opção de que seja tocado um som, enviado um e-mail e ainda a ligação remota de ventiladores ou nebulizadores. Os dados são registrados em dois gráficos, no primeiro são apresentados os valores dos índices de conforto, e no segundo gráfico, podem ser analisados os valores das variáveis utilizadas para o cálculo, por exemplo, temperatura de bulbo seco e temperatura de bulbo úmido, propiciando ao produtor fazer o acompanhamento da evolução dos índices versus as medidas aplicadas para se obter a normalização das condições de temperatura.

Com isso abriram-se várias oportunidades, que serão discutidas no Capítulo 5 na parte referente a estudos futuros.

CAPÍTULO IV

4 ANÁLISE E DISCUSSÃO DOS RESULTADOS

Após a finalização dos testes, os resultados obtidos pelo programa computacional mostraram-se compatíveis com a literatura pesquisada (KELLY e BOND, 1971; TAO; XIN, 2003; BUFFINGTON *et al.*, 1981).

4.1 Intervalos Utilizados no Programa Computacional

Devido a predominância do clima tropical no Brasil, o programa computacional foi otimizado para temperaturas entre a região de homeotermia até a região de hipertermia. Quando a temperatura do ambiente ultrapassa a TCS, os animais reagem através da dissipação de calor utilizando os mecanismos sensíveis como condução, convecção, radiação e mecanismos latentes de evaporação. Esses mecanismos dependem da existência de gradiente de temperatura entre o animal e o ambiente. Os intervalos estão definidos na Tabela 4, e a interpretação das mensagens e avisos do programa computacional encontram-se a seguir.

Conforto:

- Indica os valores dos índices nos quais os mecanismos responsáveis pela manutenção da temperatura corporal operam com um mínimo de dispêndio de energia por parte dos animais.
- O metabolismo basal funciona sem sobrecargas.
- Os processos fisiológicos atuam de maneira natural.
- Não são necessárias alterações comportamentais.

Alerta e Perigo:

- O animal aciona seus mecanismos termo regulatórios para auxiliar a dissipação do calor para o ambiente
- O metabolismo basal e os processos fisiológicos começam a sofrer sobrecarga.

- Implica no aumento de dias para chegar ao peso de abate, pela menor ingestão diária de nutrientes e piora na conversão alimentar, devido ao fato de que parte dos nutrientes consumidos passará a ser desviada para a manutenção da homeotermia, levando a prejuízos econômicos.
- Adicionalmente, nas aves temos a incidência de alcalose respiratória, que leva a morte em casos extremos, ou acarreta ovos pequenos e de casca fina, devido a reação do organismo de utilizar o cálcio para neutralizar a acidez do sangue.

Emergência:

- Hipertermia: não se consegue obter o resfriamento necessário para a manutenção do equilíbrio homeotérmico e a temperatura corporal aumenta cada vez mais.
- Devem ser tomadas medidas imediatas na busca de trazer os valores para índices normais.
- Risco iminente de morte.

4.2 Validação dos Resultados Obtidos

Os resultados obtidos foram validados através da criação aleatória de séries de dados compostas por 108 valores de temperaturas e velocidades do ar conforme recomendados pela literatura (KELLY, BOND, 1971; TAO, XIN, 2003; BUFFINGTON *et al.*, 1981). Os resultados foram também apresentados a especialistas, que reportaram que o comportamento do programa computacional se encontrava dentro do esperado. Os dados contemplaram todas as espécies de animais, no entanto, só estão apresentados nas tabelas a seguir os resultados para aves de corte.

4.2.1 ITU

Os intervalos de temperatura utilizados tiveram seus valores gerados entre 0º C e 45º C.

Os resultados obtidos através do processamento dos dados pelo programa computacional encontram-se na Tabela 5.

Tabela 5 – ITU - Valores e Status obtidos através de simulação utilizando dados hipotéticos a partir de uma série de dados aleatórios (Frangos)

Temperatura (°C)		ITU	Status	Temperatura (°C)		ITU	Status	Temperatura (°C)		ITU	Status
Bulbo Seco	Bulbo Úmido			Bulbo Seco	Bulbo Úmido			Bulbo Seco	Bulbo Úmido		
43	28	91,72	Emergência	5	18	57,16	Conforto	31	44	94,60	Emergência
7	33	69,40	Conforto	32	35	88,84	Emergência	2	14	52,12	Conforto
26	6	63,64	Conforto	33	2	65,80	Conforto	20	31	77,32	Alerta
19	9	60,76	Conforto	27	38	87,40	Emergência	7	44	77,32	Alerta
9	1	47,80	Conforto	33	37	91,00	Emergência	1	44	73,00	Conforto
12	29	70,12	Conforto	21	35	80,92	Perigo	24	14	67,96	Conforto
25	37	85,24	Emergência	27	38	87,40	Emergência	45	45	105,40	Emergência
24	15	68,68	Conforto	7	28	65,80	Conforto	15	45	83,80	Perigo
36	5	70,12	Conforto	16	22	67,96	Conforto	30	36	88,12	Emergência
36	23	83,08	Perigo	9	21	62,20	Conforto	42	39	98,92	Emergência
31	39	91,00	Emergência	9	7	52,12	Conforto	12	39	77,32	Alerta
3	27	62,20	Conforto	18	5	57,16	Conforto	6	29	65,80	Conforto
43	26	90,28	Emergência	34	31	87,40	Emergência	42	10	78,04	Alerta
3	34	67,24	Conforto	1	37	67,96	Conforto	29	2	62,92	Conforto
12	27	68,68	Conforto	20	37	81,64	Perigo	10	21	62,92	Conforto
5	13	53,56	Conforto	44	45	104,68	Emergência	25	30	80,20	Perigo
15	30	73,00	Conforto	0	38	67,96	Conforto	41	31	92,44	Emergência
2	3	44,20	Conforto	20	13	64,36	Conforto	30	41	91,72	Emergência
19	40	83,08	Perigo	14	17	62,92	Conforto	45	6	77,32	Alerta
21	17	67,96	Conforto	33	38	91,72	Emergência	26	25	77,32	Alerta
2	19	55,72	Conforto	0	2	42,04	Conforto	25	9	65,08	Conforto
2	10	49,24	Conforto	4	38	70,84	Conforto	19	41	83,80	Perigo
1	28	61,48	Conforto	28	13	70,12	Conforto	22	17	68,68	Conforto
34	44	96,76	Emergência	0	31	62,92	Conforto	37	39	95,32	Emergência
44	33	96,04	Emergência	30	40	91,00	Emergência	8	43	77,32	Alerta
11	2	49,96	Conforto	3	25	60,76	Conforto	38	19	81,64	Perigo
39	14	78,76	Alerta	24	12	66,52	Conforto	42	33	94,60	Emergência
12	41	78,76	Alerta	10	2	49,24	Conforto	44	25	90,28	Emergência
28	32	83,80	Perigo	13	23	66,52	Conforto	24	15	68,68	Conforto
29	23	78,04	Alerta	33	19	78,04	Alerta	36	41	96,04	Emergência
25	19	72,28	Conforto	41	16	81,64	Perigo	31	43	93,88	Emergência
32	18	76,60	Alerta	34	3	67,24	Conforto	26	1	60,04	Conforto
25	21	73,72	Conforto	27	15	70,84	Conforto	19	35	79,48	Perigo
32	41	93,16	Emergência	45	20	87,40	Emergência	3	43	73,72	Conforto
11	13	57,88	Conforto	32	1	64,36	Conforto	13	19	63,64	Conforto
23	31	79,48	Perigo	21	17	67,96	Conforto	19	2	55,72	Conforto

4.2.2 ITUV

Os intervalos de temperatura utilizados tiveram seus valores gerados entre 0°C e 45°C. Para a velocidade do ar o intervalo foi de 0,01 à 5,00 ms⁻¹.

Os resultados obtidos através do processamento dos dados pelo programa computacional encontram-se na Tabela 6.

Tabela 6 - ITUV - Valores e Status obtidos através de simulação utilizando dados hipotéticos a partir de uma série de dados aleatórios (Frangos)

Temperatura (°C)		Velocidade do ar (m/s)	ITUV	Status
Bulbo Seco	Bulbo Úmido			
17	13	1,00	16,40	Conforto
27	21	4,00	24,08	Alerta
9	32	5,00	11,34	Conforto
44	13	1,00	39,35	Emergência
20	10	4,00	17,07	Conforto
10	24	2,00	11,62	Conforto
7	20	4,00	8,26	Conforto
45	43	5,00	40,72	Emergência
3	12	3,00	4,08	Conforto
27	44	5,00	26,92	Alerta
15	23	4,00	14,95	Conforto
18	45	5,00	20,08	Conforto
9	12	2,00	9,08	Conforto
13	43	5,00	15,94	Conforto
8	38	2,00	12,01	Conforto
2	21	4,00	4,48	Conforto
13	9	2,00	11,91	Conforto
26	40	2,00	26,99	Alerta
6	43	5,00	10,52	Conforto
25	5	5,00	20,04	Conforto
34	17	3,00	29,51	Alerta
16	38	4,00	17,81	Conforto
35	13	4,00	29,25	Alerta
0	7	1,00	1,05	Conforto
12	0	2,00	9,80	Conforto
22	30	1,00	23,20	Conforto
12	14	2,00	11,82	Conforto
25	33	1,00	26,20	Alerta
33	16	2,00	29,25	Alerta
35	27	4,00	31,19	Alerta
39	35	4,00	35,43	Perigo
3	31	2,00	6,92	Conforto
19	29	5,00	18,67	Conforto
21	24	2,00	20,60	Conforto
41	21	4,00	35,06	Perigo
45	24	1,00	41,85	Emergência

Temperatura (°C)		Velocida de do ar (m/s)	ITUV	Status
Bulbo Seco	Bulbo Úmido			
12	45	1,00	16,95	Conforto
22	1	4,00	17,39	Conforto
12	3	5,00	9,70	Conforto
38	37	4,00	34,93	Perigo
29	1	1,00	24,80	Alerta
31	40	4,00	29,85	Alerta
28	32	5,00	26,05	Alerta
33	2	5,00	25,82	Alerta
1	9	2,00	2,11	Conforto
3	35	5,00	7,10	Conforto
34	40	3,00	32,75	Alerta
36	18	2,00	31,99	Alerta
18	22	1,00	18,60	Conforto
28	22	4,00	25,01	Alerta
34	3	4,00	27,08	Alerta
29	34	1,00	29,75	Alerta
19	35	1,00	21,40	Conforto
11	17	3,00	11,17	Conforto
4	2	3,00	3,47	Conforto
28	34	5,00	26,32	Alerta
23	32	3,00	22,85	Conforto
26	30	5,00	24,23	Alerta
36	23	2,00	32,71	Alerta
27	25	3,00	25,05	Alerta
2	38	1,00	7,40	Conforto
38	4	5,00	29,97	Alerta
27	34	4,00	25,88	Alerta
41	21	4,00	35,06	Perigo
38	14	1,00	34,40	Perigo
34	16	2,00	30,07	Alerta
31	20	2,00	28,19	Alerta
37	42	3,00	35,42	Perigo
2	0	4,00	1,57	Conforto
14	28	1,00	16,10	Conforto
20	37	3,00	21,16	Conforto
11	28	5,00	12,34	Conforto

Temperatura (°C)		Velocida de do ar (m/s)	ITUV	Status
Bulbo Seco	Bulbo Úmido			
25	23	1,00	24,70	Alerta
43	30	3,00	38,52	Perigo
5	25	1,00	8,00	Conforto
33	4	2,00	27,52	Alerta
36	36	2,00	34,58	Perigo
22	39	5,00	22,36	Conforto
13	5	2,00	11,34	Conforto
6	7	4,00	5,67	Conforto
5	14	3,00	5,96	Conforto
22	42	3,00	23,46	Conforto
2	14	4,00	3,51	Conforto
30	36	2,00	29,68	Alerta
14	15	2,00	13,59	Conforto
41	31	5,00	35,98	Perigo
8	23	1,00	10,25	Conforto
18	42	5,00	19,67	Conforto
24	25	4,00	22,28	Conforto
45	33	4,00	39,86	Emergência
3	3	4,00	2,77	Conforto
4	19	5,00	5,69	Conforto
45	26	5,00	38,39	Perigo
42	3	1,00	36,15	Perigo
0	12	2,00	1,73	Conforto
33	23	5,00	28,69	Alerta
0	16	3,00	2,25	Conforto
15	28	5,00	15,44	Conforto
0	27	1,00	4,05	Conforto
44	30	3,00	39,31	Emergência
4	32	4,00	7,57	Conforto
24	5	4,00	19,52	Conforto
28	41	2,00	28,77	Alerta
14	12	5,00	12,48	Conforto
5	29	1,00	8,60	Conforto
27	8	5,00	22,00	Conforto
32	6	1,00	28,10	Alerta
0	23	4,00	3,18	Conforto

4.2.3 IGNU

Os intervalos de temperatura utilizados tiveram seus valores gerados entre 0°C e 45° C.

Os resultados obtidos através do processamento dos dados pelo programa computacional encontram-se na Tabela 7.

Tabela 7 - IGNU - Valores e Status obtidos através de simulação utilizando dados hipotéticos a partir de uma série de dados aleatórios (Frangos)

Temperatura (°C)		IGNU	Status	Temperatura (°C)		IGNU	Status	Temperatura (°C)		IGNU	Status
Globo Negro	Ponto de Orvalho			Globo	Ponto de Orvalho			Globo	Ponto de Orvalho		
42	8	69,58	Conforto	12	26	58,06	Conforto	35	19	69,34	Conforto
6	14	50,14	Conforto	24	23	64,18	Conforto	17	16	57,46	Conforto
16	22	59,02	Conforto	23	35	67,90	Conforto	10	9	50,74	Conforto
21	23	62,38	Conforto	23	28	65,38	Conforto	19	11	56,86	Conforto
6	44	60,94	Conforto	26	13	61,78	Conforto	25	7	59,02	Conforto
37	45	79,90	Emergência	21	16	59,86	Conforto	5	18	50,98	Conforto
19	25	61,90	Conforto	39	37	78,22	Emergência	45	20	75,70	Conforto
19	43	68,38	Conforto	18	43	67,78	Conforto	4	26	53,26	Conforto
25	35	69,10	Conforto	6	31	56,26	Conforto	37	1	64,06	Conforto
1	2	42,82	Conforto	12	12	53,02	Conforto	45	37	81,82	Emergência
7	26	55,06	Conforto	19	3	53,98	Conforto	19	25	61,90	Conforto
20	35	66,10	Conforto	13	19	56,14	Conforto	44	24	76,54	Emergência
14	42	65,02	Conforto	15	31	61,66	Conforto	1	9	45,34	Conforto
45	17	74,62	Conforto	11	27	57,82	Conforto	39	8	67,78	Conforto
11	36	61,06	Conforto	33	38	74,98	Conforto	3	28	53,38	Conforto
15	41	65,26	Conforto	4	30	54,70	Conforto	2	40	57,10	Conforto
15	34	62,74	Conforto	17	22	59,62	Conforto	41	32	77,62	Emergência
26	37	70,42	Conforto	32	34	72,94	Conforto	27	28	67,78	Conforto
27	26	67,06	Conforto	31	12	64,42	Conforto	20	42	68,62	Conforto
12	0	48,70	Conforto	41	4	67,54	Conforto	18	16	58,06	Conforto
36	22	71,02	Conforto	13	29	59,74	Conforto	24	41	70,66	Conforto
24	20	63,10	Conforto	33	39	75,34	Conforto	5	18	50,98	Conforto
40	29	75,94	Conforto	37	20	70,90	Conforto	36	21	70,66	Conforto
25	22	64,42	Conforto	27	40	72,10	Conforto	32	43	76,18	Emergência
11	26	57,46	Conforto	8	8	49,18	Conforto	32	36	73,66	Conforto
35	45	78,70	Emergência	16	38	64,78	Conforto	7	7	48,22	Conforto
42	21	74,26	Conforto	11	38	61,78	Conforto	42	38	80,38	Emergência
23	25	64,30	Conforto	25	11	60,46	Conforto	24	35	68,50	Conforto
33	2	62,02	Conforto	21	28	64,18	Conforto	16	24	59,74	Conforto
12	14	53,74	Conforto	31	41	74,86	Conforto	17	29	62,14	Conforto
15	15	55,90	Conforto	21	1	54,46	Conforto	13	44	65,14	Conforto
31	1	60,46	Conforto	7	10	49,30	Conforto	44	44	83,74	Emergência

4.3 Exemplos da Utilização do Programa Computacional

Selecionou-se o opção Avicultura e o Índice ITU. A seguir foram digitados dados nos campos temperatura de bulbo seco variando de 18,00 °C a 27,00 °C, e para a temperatura de bulbo úmido os valores variaram de 15,00 °C a 19 °C. Os ITU obtidos oscilaram de 64,36 a 73,72. Esses valores correspondem ao intervalo de conforto (Tabela 4).

O programa computacional correspondeu a essas entradas de maneira correta, os dados foram exibidos nos gráficos correspondentes, a cor selecionada foi o verde, as mensagens informaram a normalidade das temperaturas e não foi solicitada nenhuma providência do usuário. Não houve o acionamento dos equipamentos remotos (ventiladores e nebulizadores). Ao solicitar o primeiro cálculo, devido ao fato de estarem selecionadas as opções habilitar sons e enviar e-mails, o programa respondeu da maneira esperada, executando essas duas ações.

Os resultados do processamento encontram-se exibidos na Figura 19, o e-mail enviado pelo programa computacional e os dados são mostrados na Figura 20.



Figura 19 - Dados obtidos para o cálculo do ITU dentro da faixa Normal

Status: Alerta
Data: 10/06/2013 22:51:48
Valor do ITU: 76,60
Mensagem: São necessárias medidas para a diminuição da temperatura

Você está recebendo esse e-mail por estar cadastrado na lista de usuários do Sistema de Controle dos Índices de Conforto Térmico. Em caso de dúvida contate o administrador do sistema. Obrigado.

Figura 20 - E-mail enviado pelo sistema ao atingir o ITU dentro da faixa de conforto

Para temperaturas de bulbo seco variando de 28,00 °C a 32,00 °C, e temperaturas de bulbo úmido de 19,00 °C a 21 °C. Os ITU obtidos oscilaram de 74,44 a 78,76. Esses valores correspondem ao intervalo de alerta (Tabela 4).

Nessa situação, a cor do gráfico e das mensagens é alterada para amarelo, o programa computacional emite um som de alerta, informa e solicita intervenção do usuário, envia um e-mail para os destinatários cadastrados, ativa o ventilador e o nebulizador.

A verificação do processamento pode ser visualizado na Figura 21, e o e-mail enviado é mostrado na Figura 22.

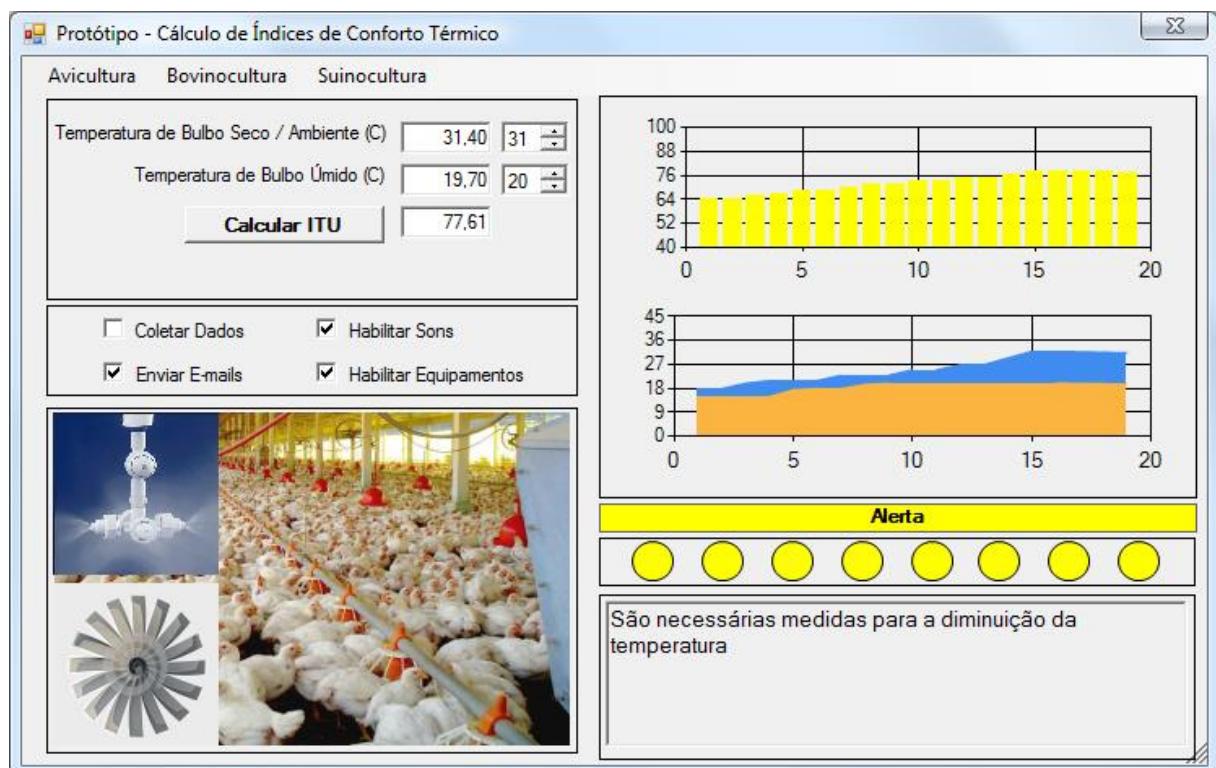


Figura 21 - Dados obtidos para o cálculo do ITU dentro da faixa de Alerta

Status: Alerta
Data: 10/06/2013 22:51:48
Valor do ITU: 76,60
Mensagem: São necessárias medidas para a diminuição da temperatura

Você está recebendo esse e-mail por estar cadastrado na lista de usuários do Sistema de Controle dos Índices de Conforto Térmico. Em caso de dúvida contate o administrador do sistema. Obrigado.

Figura 22 - E-mail enviado pelo sistema ao atingir o ITU dentro da faixa de atenção

Com o acionamento do ventilador e do nebulizador, os valores das temperaturas iniciaram um processo de baixa, e ao atingirem 27,20 °C para bulbo seco e 18,60 °C para bulbo úmido, o ITU resultante foi de 73,79.

Adentrando assim a faixa de conforto (Tabela 4), a cor das mensagens e do gráfico do ITU foi alterada para verde, foi emitido um som de alerta, e a opção de enviar e-mails não estava habilitada. O ventilador e o nebulizador foram desligados (Figura 23).



Figura 23 - Retorno ao estado de conforto devido ao acionamento dos equipamentos (Alerta)

Para temperaturas de bulbo seco variando de 33,00 °C a 36,00 °C, e temperaturas de bulbo úmido de 21,00 °C a 24 °C. Os ITU obtidos oscilaram de 83,60 a 78,76. Esses valores correspondem ao intervalo de emergência (Tabela 4).

Nessa situação, a cor do gráfico e das mensagens é alterada para vermelho, o programa computacional emite um som de alerta, informa e solicita intervenção do usuário, não é enviado um e-mail devido à opção não estar habilitada. Ativa o ventilador e o nebulizador em um nível de maior intensidade. Caso estejam disponíveis, um número maior de equipamentos será acionado.

A verificação do processamento pode ser visualizado na Figura 24, e o e-mail enviado é mostrado na Figura 25.

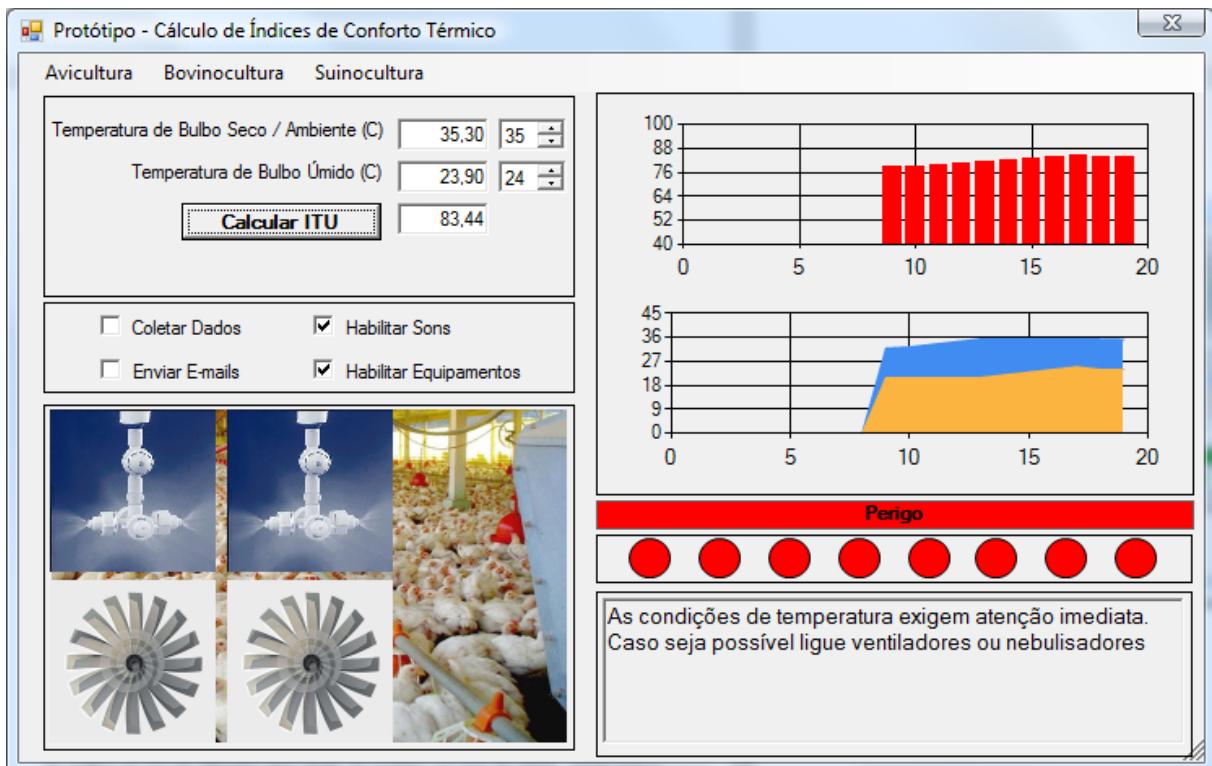


Figura 24- Dados obtidos para o cálculo do ITU dentro da faixa de Perigo

Status: Perigo

Data: 10/06/2013 23:00:10

Valor do ITU: 83,80

Mensagem: As condições de temperatura exigem atenção imediata. Caso seja possível ligue ventiladores ou nebulizadores

Você está recebendo esse e-mail por estar cadastrado na lista de usuários do Sistema de Controle dos Índices de Conforto Térmico. Em caso de dúvida contate o administrador do sistema. Obrigado.

Figura 25 - E-mail enviado pelo sistema ao atingir o ITU dentro da faixa de Perigo

Após o acionamento do(s) ventilador(es) e nebulizador(es), os valores das temperaturas iniciaram um processo de baixa, sendo que ao atingirem 26,50 °C para bulbo seco e 19,50 °C para bulbo úmido, o ITU resultante foi de 73,72.

Ao ser atingida a faixa de conforto (Tabela 4). A cor das mensagens e do gráfico do ITU foi alterada para verde, foi emitido um som de alerta, sendo que a opção de enviar e-mails não estava habilitada. O(s) ventilador(es) e nebulizador(es) foram desligados (Figura 26).



Figura 26 - Retorno ao estado de conforto devido ao acionamento dos equipamentos (Perigo)

Para temperaturas de bulbo seco acima de 36,00 °C, e as temperaturas de bulbo úmido a partir de 24 °C, os valores de ITU obtidos ficam acima dos 84,16 correspondendo ao intervalo de perigo (Tabela 4).

Nessa situação, a cor do gráfico e das mensagens é alterada para preto, o programa computacional emite um som de alerta, informa e solicita intervenção do usuário e envia um e-mail para os destinatários cadastrados. Ativa o ventilador e o nebulizador em seus níveis máximos. Caso estejam disponíveis, todos os equipamentos remotos serão acionados.

A verificação do processamento pode ser visualizado na Figura 27, e o e-mail enviado é mostrado na Figura 28.

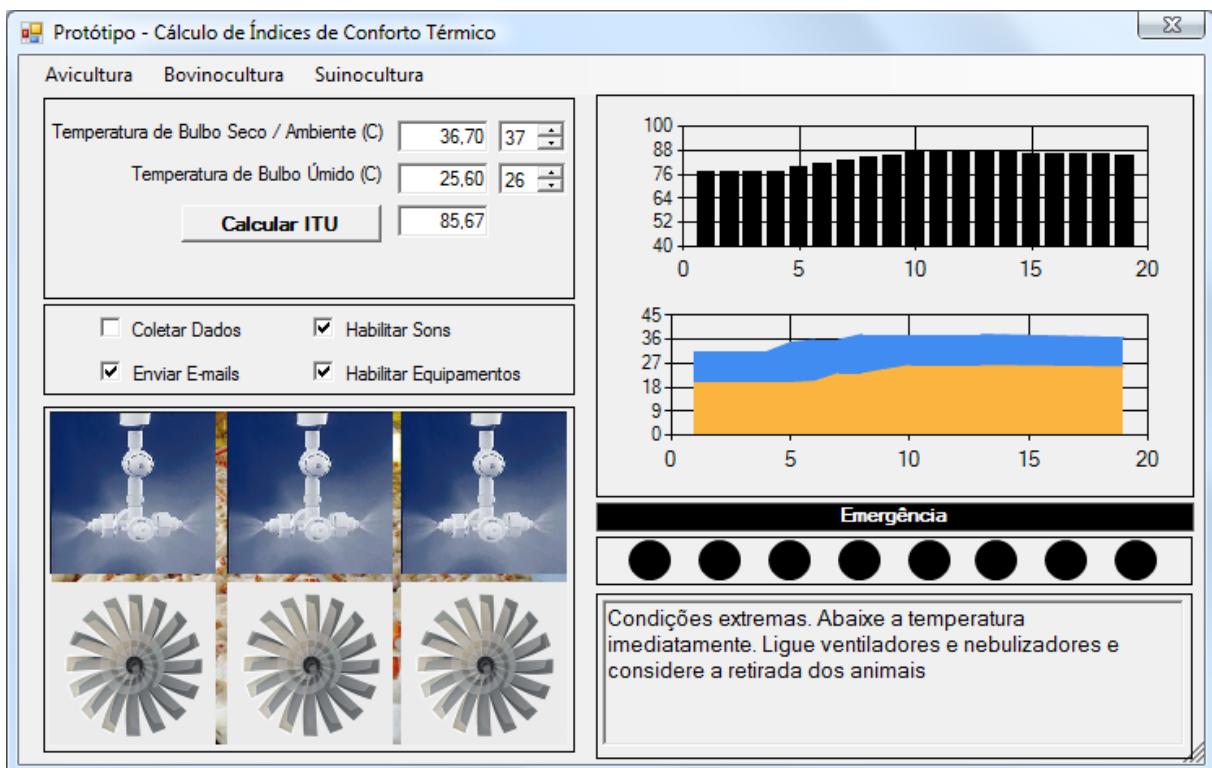


Figura 27 - Dados obtidos para o cálculo do ITU dentro da faixa de Emergência

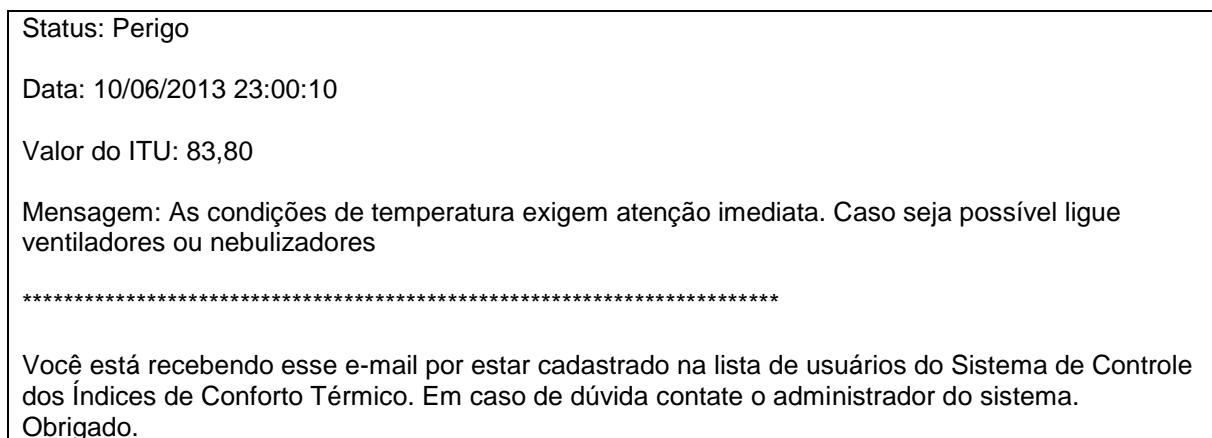


Figura 28- E-mail enviado pelo sistema ao atingir o ITU dentro da faixa de Emergência

Após o acionamento do(s) ventilador(es) e nebulizador(es), os valores das temperaturas iniciaram um processo de baixa, e ao atingirem 24,10 °C para bulbo seco e 21,80 °C para bulbo úmido, o ITU resultante foi de 73,65.

Ao ser atingida a faixa de conforto (Tabela 4), a cor das mensagens e do gráfico do ITU foi alterada para verde, foi emitido um som de alerta, e a opção de enviar emails não estava habilitada. O(s) ventilador(es) e nebulizador(es) foram desligados (Figura 29).



Figura 29 - Retorno ao estado de conforto devido ao acionamento dos equipamentos (Emergência)

De maneira análoga, os procedimentos acima descritos foram repetidos para todos os índices contemplados pelo programa computacional:

ITU: Avicultura, bovinocultura e suinocultura;

ITUV: Avicultura;

IGNU: Avicultura, bovinocultura e suinocultura.

Os resultados foram satisfatórios em todos os casos. Com isso verificou-se que o programa atende aos requisitos, estando em conformidade com os algoritmos e os intervalos obtidos através de pesquisa bibliográfica (Tabela 2, 3 e 4).

CAPÍTULO V

5 CONCLUSÃO

A partir dos algoritmos selecionados, foi possível desenvolver um programa computacional para calcular os índices de conforto térmico para frangos, suínos e bovinos de leite em ambientes de produção intensiva.

Através de pesquisa bibliográfica, foi feita a interpretação desses índices, distribuindo os resultados em intervalos denominados conforto, alerta, emergência e perigo.

Utilizando-se a Engenharia de Software, efetuou-se a especificação, análise e codificação do programa computacional que atendesse aos requisitos descritos no Capítulo 2.

Procedeu-se a validação do programa computacional através de séries de dados geradas dentro de intervalos pré-estabelecidos pelos especialistas, sendo os resultados obtidos confrontados com valores descritos na literatura e analisados por especialistas da área.

Com isso mostrou-se viável desenvolver um programa computacional que possa identificar situações ótimas de alojamento em relação ao conforto térmico, através de poucas variáveis de fácil obtenção no ambiente de produção para a produção industrial de frangos, suínos e bovinos de leite, sugerindo ações ao produtor para a mitigação dos riscos.

Considerando-se a importância do mercado de carnes para o Brasil, tanto em termos de quantidades como de valores, pode-se inferir que incrementos na produtividade, como por exemplo, uma taxa menor de mortandade, ou, menor quantidade de alimentos necessários para a obtenção de taxas ótimas de produção, signifiquem maiores lucros para toda a cadeia de produção.

Nesse panorama, ferramentas como o programa computacional aqui desenvolvido se tornam parte importante desse processo, permitindo um maior controle do ambiente da produção.

5.1 Propostas para Estudos Futuros

Criação de um banco de sugestões, que possa ser alimentado por atualizações de *software* ou também pelo produtor, e após a validação por especialistas, seria compartilhado entre todos os usuários, agregando valor à cadeia de produção e contribuindo para que sejam alcançados nos objetivos de aumento de produtividade e manutenção dos índices de conforto dos animais.

Identificação de outros indicadores que sejam importantes para o processo de produção e que sejam passíveis de serem integrados ao programa computacional.

Utilização de redes neurais na interpretação dos índices de conforto.

Transformação do programa computacional em um programa especialista, que passe a agir de maneira pró ativa ao ser utilizado pelo produtor.

REFERÊNCIAS BIBLIOGRÁFICAS

- ABREU, P. G. *et al.* Utilização de piso aquecido eletricamente na criação de aves. **Engenharia na Agricultura**, Viçosa, v. 4, n. 12, p. 1-19, 1995.
- ADELHARD, K. *et al.* A prototype of a computerized patient record. **Computer methods and programs in biomedicine**, v.48, p.115-9, set-out. 1995.
- ABRAN, A., ROBILLARD, P.N. Function Points Analysis: An Empirical Study of its Measurement Processes. **IEEE Transactions on Software Engineering**, v.22, n.12, 1996.
- BACCARI JUNIOR, F. Adaptação de sistemas de manejo na produção de leite em climas quentes. In: **SIMPÓSIO BRASILEIRO DE AMBIÊNCIA NA PRODUÇÃO DE LEITE**, Piracicaba, 1998. Anais... Piracicaba: FEALQ, 1998, p.24-67.
- BAÊTA, F.C. Responses of lactating dairy cows to the combined effects of temperature humidity and wind velocity in the warm season. 1985. 218 f. Tese Doutorado – University of Missouri, Columbia, 1985.
- BAÊTA, F.C., SOUZA, C. F. **Ambiência em edificações rurais – conforto animal**. Ed. UFV, Viçosa – MG, 1997. p. 246
- BAUER, F.L. **Software Engineering - Information Processing**. Amsterdam: North Holland Publishing, 1972
- BOEHM, B.W. **Software Engineering Economics**. Englewood Cliffs: Prentice-Hall, 1981.
- BROWN-BRANDL, T. *et al.* **Temperature humidity index for growing tom turkeys**. **Transactions of the ASAE**, v.40, n.1, p.203-209, 1997.
- BUFFINGTON, D *et al.* Black globe-humidity index (BGHI) as comfort equation for dairy cows. **Transactions of the ASAE**, v.24, n.3, p.711-14, 1981.
- CELLA, P. e outros. Níveis de lisina mantendo a relação aminoacídica para frangos de corte no período de 1 a 21 dias de idade, em diferentes ambientes térmicos. **Revista Brasileira de Zootecnia**, Viçosa, v.30, n.2, p.440- 448, 2001
- COTIS. Consórcio de Tecnologia da Informação na Área da Saúde. Capturado em 25/06/2001. Online. Disponível na Internet: <http://www.cotis.com.br>.
- DAVIS, A.M. Software Life Cycle Models. In: Thayer, R. (ed.). **Software Engineering Project Management. Second Edition**. Los Alamitos: IEEE, 1997.

DAVIS, M.W. **Computerizing Healthcare Information: Developing Electronic Patient Information Systems. Revised edition.** New York: McGraw-Hill, 1998.

DEGOULET, P., FIESCHI, M. **Introduction to Clinical Informatics.** New York: Springer-Verlag, 1997.

DOLIN, R.H. A high-level object-oriented model for representing relationship in an electronic medical record. **Proceedings of the Annual Symposium on Computer Application in Medical Care**, p.514-8, 1994.

DORFMAN, M. **Requirements Engineering.** In: Thayer, R., Dorfman, M. (eds.). **Software Requirements Engineering.** Los Alamitos: IEEE, 1997.

FALCO, J. E. **Bioclimatologia Animal.** Lavras: UFLA, 1997. 57 p.

FIORELLI, Juliano *et al.* Avaliação da eficiência térmica de telha reciclada à base de embalagens longa vida. **Revista Brasileira de Engenharia Agrícola e Ambiental** v.13, n.2, p.204–209, 2009.

FURLAN, J.D. **Modelagem de objetos através da UML.** São Paulo: Makron Books, 1998.

FUZION. **Introdução a Orientação a Objetos.** Rio de Janeiro, 1999. CD-ROM. E-book.

GAMMA, E. *et al.* **Design Patterns.** Addison-Wesley, 1995.

GLINZ, M. Problems and Deficiencies of UML as a Requirements Specification Language. **Proceedings of the 10th International Workshop on Software Specification and Design** (IWSSD-10), p.11-22, 2000.

JACOBSON, I. *et al.* **Object-Oriented Software Engineering: A Use Case Driven Approach.** Addison-Wesley, 1996.

KELLY, C.F.; BOND, T.E. **Bioclimatic factors and their measurements.** In: National Academy Of Sciences, (Ed.) **A guide to environmental research on animals. Washington:** National Academy of Sciences, 1971. p.71-92.

KUIKKA, E., EEROLA, A., PORRASMAA, J. *et al.* **Design of the SGML-based electronic patient record system with the use of object-oriented analysis methods.** In: P. Kokol *et al.* (eds.): **MIE-99 - Fifteenth International Congress of European Federation for Medical Informatics.** Amsterdam: IOS Press, 1999. p.838-841.

LEÃO, B.F., BANDARRA, E.B., ARAÚJO, L.F., MADRIL, P.J., MOURA, L., SIGULEM, D. CCS-SIS – **O Consórcio de Componentes de Software para Sistemas de Informação em Saúde**. Anais do IV FNCTS - Fórum Nacional de Ciência e Tecnologia em Saúde. Curitiba. 1998.

MARTIN, J., ODELL, J.J. **Object-Oriented Analysis and Design**. Prentice-Hall Inc., 1992

MATTIAZZI, L.D. **Orientação a Objetos e a UML: Finalmente um Rumo a Seguir**. Developers' Magazine, Rio de Janeiro, ano III, n.26, p.26-29, jul. 1998.

MILHOLLAND, D.K. **Information systems in critical care: a measure of their effectiveness**. Proceedings Medinfo, p.1068-70, 1995.

MURPHY, G.F., HANKEN, M.A., WATERS, K.A. **Electronic Health Records: Changing the Vision**. Philadelphia: W.B. Saunders Company, 1999.

NÄÄS, I. *et al.* Efeito das temperaturas de bulbo seco e de globo negro e do índice de temperatura e umidade, em vacas em produção alojadas em sistema de free-stall. **Brazilian Journal of Veterinary Research and Animal Science**. v. 39, n. 6, p. 320-323, 2002.

NÄÄS, I. A.; MOURA, D. J.; LAGANÁ, C. A. A amplitude térmica e seu reflexo na produtividade de frangos de corte. In: **CONFERÊNCIA APINCO DE CIÊNCIA E TECNOLOGIA AVÍCOLAS, 1995**. Anais... Campinas, Facta, p. 203-204.

OLIVEIRA, P. A. V. Efeito do tipo de telha sobre o acondicionamento ambiental e o desempenho de frangos de corte. In: **CONFERÊNCIA APINCO DE CIÊNCIA E TECNOLOGIA AVÍCOLAS, 1995**. Anais... Campinas, Facta. p. 297- 298.

PERREAU, L.E., WIEDERHOLD, G. **System Design and Evaluation**. In: Shortliffe, E.H., Fagan, L.M. (eds). Medical Informatics: Computer Applications in Health Care. New York: Addison-Wesley Publishing, 1990.

PRESSMAN, R.S. **Engenharia de Software**. São Paulo: Makron Books, 1995

RAMAMOORTHY, C.V. *et al.* **Software Engineering: problems and perspectives**. Computer. p.191-209, out. 1984.

RUMBAUGH, J. *et al.* **Modelagem e projetos baseados em objetos**. Rio de Janeiro: Editora Campus, 1994

RUTZ, F. Aspectos fisiológicos que regulam o conforto térmico das aves. In: **CONFERÊNCIA APINCO 1994 DE CIÊNCIA E TECNOLOGIA AVÍCOLAS**, 1994, São Paulo. Anais..., São Paulo: FACTA, 1994. p.99-136.

SAMPAIO, C. A. *et al.* Avaliação do ambiente térmico em instalação para crescimento e terminação de suínos utilizando os índices de conforto térmico nas condições tropicais. **Ciência Rural**, maio- junho, ano/vol. 34, número 003, 2004.

SANTOS, A. C.; BAÊTA, F. C.; CECON, P. R.; CARDOSO, R. M. Análise de diferentes bezerreiros individuais móveis, para região de Viçosa. **Engenharia na Agricultura**, Viçosa, v. 2, n. 7, p. 1-8, 1993.

SANTOS, P. *et al.* Ambiente térmico no interior de galpões avícolas em escalas reduzida com ventilação natural e artificial dos telhados. **Eng. Agrícola**, Jaboticabal, v.25, n.3, p.575-584, set./dez. 2005

SAKAMOTO, N. **A practical object-oriented approach to a development of a next generation hospital information systems**. Proceedings Medinfo, p.957-61, 1998.

SCHLOEFFEL, P. GEHR - The Good Electronic Health Record. 1998. Capturado em 30/05/2001. Online. Disponível na Internet <http://www.gehr.org/Documents/Peter%20Schloeffel%20GEHRNZMoH.rtf>.

SEI. **The Capability Maturity Model guidelines for improving the software process**. Adisson-Wesley, 1995.

SETO, K., KAMIYAMA, T., MATSUO, H. **An object-modeling method for hospital information systems**. Proceedings Medinfo, p.981-5, 1998.

SIEGEL, H. S. 1968. **Adaptation of poultry**. In: Adaptation of domestic animals, Lea & Febiger, Philadelphia.

STEINBACH, J. 1971. **Relative effects of temperature and humidity on thermal comfort in swine**. Nigeria Agricultural Journal. 8: 132-134.

SUTTER, H. **The Free Lunch Is Over: A Fundamental Turn Toward Concurrency in Software**, 2009.

TAO, X.; XIN, H. Acute synergistic effects of air temperature, humidity, and velocity on homeostasis of market-size broilers. **Transactions of the ASAE**, St. Joseph, v. 46, n. 2, p. 491-497, Mar./Abr. 2003.

TEIXEIRA, V.H. **Estudo dos índices de conforto em duas instalações de frango de corte para as regiões de Viçosa e Visconde do Rio Branco - MG**. Viçosa, 1983. 62p. Dissertação (Mestrado) – Universidade Federal de Viçosa.

TINOCO, I. F. F. Ambiência e instalações para a avicultura industrial. In: **27º Congresso Brasileiro De Engenharia Agrícola**, 1998, Poços de Caldas. Lavras: UFLA/SBEA, 1998, p.1-86.

TURCO, S. H. N.; BAÊTA, F. C.; COSTA, P. M. A.; CARDOSO, R. M.; CECON, P. R. Modificações das condições ambientais de verão em maternidades de suínos. **Engenharia na Agricultura**, Viçosa, v. 3, n. 11, p. 1-12, 1994.

VAZ, R. **Rumo ao Nível II da Capability Maturity Model - CMM. Developers' Magazine**, Rio de Janeiro, ano 5, n.49, p.20-23, set. 2000.

VITORASSO, Guilherme, PEREIRA, D. Análise comparativa do ambiente de aviários de postura com diferentes sistemas de acondicionamento. **Revista Brasileira de Engenharia Agrícola e Ambiental** v.13, n.6, p.788–794, 2009.

VON MAYRHAUSER, A. **Software Engineering: Methods and Management**. San Diego: Academic Press, 1990.