

UNIVERSIDADE FEDERAL DO RIO DE JANEIRO
ESCOLA POLITÉCNICA
DEPARTAMENTO DE ELETRÔNICA E DE COMPUTAÇÃO

**SISTEMA WEB PARA AQUISIÇÃO DE MEDIDAS
REMOTAS E GESTÃO DE INFORMAÇÕES DE
PROJETOS DE PESQUISA**

Autor:

Marcelo Caetano Grael Tepedino

Orientador:

Prof. Marcelo Martins Werneck, Ph. D.

Orientador:

Cesar Cosenza de Carvalho, D. Sc.

Examinador:

Prof. Heraldo Luís Silveira de Almeida, D. Sc.

Examinador:

Eduardo Shigueo Yugue, M. Sc.

DEL

Julho de 2009

Agradecimentos

Primeiramente, gostaria de agradecer aos meus pais, Caetano e Maysa, por tudo que fazem pensando sempre no meu futuro e na realização dos meus sonhos. Obrigado pela educação e pelas oportunidades. Agradeço também ao carinho e confiança de minhas irmãs, Cynthia e Carina.

À minha namorada, Ju, pela compreensão e principalmente pelo apoio. Obrigado por estar sempre ao meu lado, me incentivando a buscar meus objetivos.

Aos amigos Fernando Ferreira e Rafael Pantoja, por tudo que passamos ao longo da faculdade juntos. Fernando, obrigado por tornar este projeto realidade em tão pouco tempo.

Aos meus orientadores Cesar Cosenza e Eduardo Shigueo pela oportunidade e conhecimento transmitido durante a realização do projeto final.

Aos amigos conquistados ao longo da faculdade, com os quais aprendi muito e que nunca serão esquecidos.

Resumo

O projeto em questão foi desenvolvido no Laboratório de Instrumentação e Fotônica do Programa de Pós-Graduação e Pesquisa de Engenharia (LIF – COPPE/UFRJ) e teve como objetivo o desenvolvimento de um sistema *web* para auxiliar a gestão dos dados no servidor de arquivos do LIF, e a aquisição dos parâmetros medidos pelos protótipos de *hardware* de campo.

Este trabalho sugere uma solução à necessidade de se estabelecer um padrão para o gerenciamento dos dados dos diferentes projetos, em virtude de não serem interligados, na maioria dos casos. Isso permitirá o fácil acesso a todos os dados aos mais diferentes níveis de usuário, dentro e fora do laboratório. Assim, a importância deste trabalho está relacionada com a possibilidade de acesso, manipulação e análise de dados provenientes de vários projetos do laboratório em um único sistema.

Para tal, foi proposto um modelo de banco de dados capaz de identificar vários tipos de parâmetros como: voltagem, temperatura, potência, entre outros, e capaz de mapear todos os canais de medição dos protótipos de *hardware* de campo.

O sistema obtém os dados através de conexões TCP/IP com os protótipos, que agem como servidores no modo passivo, utilizando o protocolo FTP. Um protocolo de transferência de dados bilateral foi elaborado para que haja a padronização dos arquivos de dados enviados e recebidos pelo laboratório, possibilitando o tratamento correto dos dados. Mensagens de erro podem ser obtidas pela falha na conexão com determinado equipamento, falhas na obtenção de medidas, falhas de *hardware*, entre outras, facilitando o processo de identificação dos equipamentos danificados.

A variação de grandezas monitoradas ao longo do tempo pode ser analisada pelo usuário por meio de tabelas ou gráficos. Estes dados podem ser exportados em arquivos no formato CSV para análises mais detalhadas. Este sistema também permite a inclusão, modificação e exclusão de projetos, protótipos, canais de medição e tipos de parâmetros do banco de dados.

Palavras-chave: MVC, CSV, FTP, banco de dados, sistema *web*, protótipo de *hardware*.

Abstract

This project was developed in the Photonics and Instrumentation Laboratory (LIF) of the Post-Graduation and Research in Engineering Program of the Federal University of Rio de Janeiro (COPPE/UFRJ), in Brazil. The objective is the development of a web-based system to assist the management of the hardware prototypes of the laboratory as well as the acquisition of the parameters measured by these prototypes.

This work establishes a standard procedure to manage data from projects developed at the laboratory, considering that in most cases they are not interlinked. This work also enables easy access to data, to different levels of users, inside and outside the Lab. Thus, the importance of this work is related with the possibility of data access, manipulation and analysis of all projects in a single system.

To achieve this, a generic database model is proposed, capable of storing, mapping and managing projects, equipments, data acquisition channels and measurements obtained from them, such as: voltage, temperature, and power, among others.

The system obtains data through TCP/IP connections with the prototypes, which have built-in passive FTP servers. A transfer protocol was developed to standardize communications between the prototypes and the Lab, enabling the correct treatment of the data. Error messages can be generated due to equipment/connection failures and faulty measurements, helping the identification of damaged and malfunctioning equipments.

The time varying data and monitored parameters can be analyzed through tables and graphs. For more detailed analyses, this data can be downloaded as CSV files. This system also allows the inclusion, modification and exclusion of entries such as projects, prototypes, data acquisition channels and measurement types.

Siglas

ACID - Atomicidade, Consistência, Isolamento e Durabilidade

AJAX - *Asynchronous JavaScript And XML*

API - *Application Programming Interface*

CSS - *Cascading Style Sheets*

CSV - *Comma-Separated Values*

DOM - *Document Object Model*

DNS - *Domain Name System*

FTP - *File Transfer Protocol*

GSM - *Global System for Mobile communications*

HTML - *Hypertext Markup Language*

IP - *Internet Protocol*

LIF - Laboratório de Instrumentação e Fotônica

MVC - *Model-View-Controller*

PHP - *PHP: Hypertext Preprocessor*

POO - Programação Orientada a Objetos

SGBD - Sistema Gerenciador de Bancos de Dados

SI - Sistema de Informação

SQL - *Structured Query Language*

TCP - *Transmission Control Protocol*

UFRJ - Universidade Federal do Rio de Janeiro

UR - Unidade Remota

W3C - *World Wide Web Consortium*

XML - *Extensible Markup Language*

Índice

1.2 -Tema.....	1
1.3 -Justificativa.....	1
1.4 -Objetivos.....	2
1.5 -Metodologia.....	3
1.6 -Descrição.....	4
1.8 -Análise do Problema.....	5
1.9 -Análise dos Requisitos.....	6
1.10 -Sistema Proposto.....	7
1.10.1 -Modelagem de dados.....	7
1.10.2 -Casos de Uso.....	9
1.12 -Banco de Dados (PostgreSQL).....	13
1.13 -Servidor Web (Apache HTTP Server).....	14
1.14 -Padrão de Arquitetura de Software MVC (Model-View-Controller).....	14
1.14.1 -Camada de modelo (Model):.....	15
1.14.2 -Camada de controle (Controller):.....	15
1.14.3 -Camada de interface (View):.....	16
1.15 -Linguagem de Programação PHP (PHP: Hypertext Preprocessor).....	17
1.16 -Framework CakePHP.....	18
1.17 -Modelo de execução AJAX (Asynchronous Javascript And XML).....	19
1.17.1 -Folhas de Estilo CSS (Cascading Style Sheets).....	20
1.17.2 -XML (Extensible Markup Language).....	20
1.17.3 -JavaScript.....	21
1.17.4 -DOM (Document Object Model).....	21
1.17.5 -XMLHttpRequest.....	22
1.18 -CSV (Comma-Separated Values).....	22
1.20 -Protocolo de Transferência de Dados.....	24
1.20.1 -Diretório de Interface Cliente/Equipamento.....	25
1.20.2 -Formato do Arquivo de Dados.....	27
1.20.3 -Tipos de Frames de Dados.....	28
1.20.4 -Exemplos.....	29

<u>1.21 -Desenvolvimento.....</u>	<u>32</u>
<u>1.21.1 -Introdução.....</u>	<u>32</u>
<u>1.21.2 -Organização das Classes.....</u>	<u>32</u>
<u>1.22 -Páginas de Interface com o Usuário.....</u>	<u>35</u>
<u>1.22.1 -Projetos.....</u>	<u>36</u>
<u>1.22.2 -Tipos de Medida.....</u>	<u>38</u>
<u>1.22.3 -Equipamentos.....</u>	<u>40</u>
<u>1.22.4 -Canais.....</u>	<u>45</u>
<u>1.22.5 -Medidas.....</u>	<u>51</u>

Lista de Figuras

Figura 1-1: Interações entre os componentes do sistema.....	3
Figura 2-2: Modelo de banco de dados para o projeto.....	8
Figura 2-3: Diagrama de casos de uso.....	9
Figura 3-4: Requisição básica de MVC.....	16
Figura 3-5: Comparação do modelo tradicional de execução com o modelo AJAX.....	20
Figura 3-6: Documento HTML em estrutura de árvore.....	22
Figura 4-7: Conexão entre o servidor do LIF, o usuário do sistema e a unidade remota.....	24
Figura 4-8: Troca de arquivos entre o servidor e uma unidade remota.....	25
Figura 4-9: Modelo MVC para a tabela de equipamentos, com suas funcionalidades.....	33
Figura 4-10: Diagrama de seqüência para acesso ao banco de dados.....	34
Figura 4-11: Página inicial do sistema.....	35
Figura 4-12: Lista de projetos do banco de dados.....	36
Figura 4-13: Formulário de edição de um projeto.....	37
Figura 4-14: Página de visualização completa de um projeto.....	37
Figura 4-15: Confirmação de exclusão de um projeto.....	38
Figura 4-16: Lista de tipos de medidas.....	38
Figura 4-17: Formulário de edição de um tipo.....	39
Figura 4-18: Página de visualização completa de um tipo.....	39
Figura 4-19: Confirmação de exclusão de um tipo.....	40
Figura 4-20: Lista de equipamentos.....	41
Figura 4-21: Página de visualização completa de um equipamento.....	41
Figura 4-22: Formulário de criação de um equipamento.....	42
Figura 4-23: Confirmação de exclusão de um equipamento.....	43
Figura 4-24: Download de arquivo, no formato CSV, de um equipamento.....	43
Figura 4-25: Mensagem de erro na conexão com o equipamento.....	44
Figura 4-26: Mensagem de sucesso no envio de solicitação de leitura.....	45
Figura 4-27: Lista de canais.....	45
Figura 4-28: Formulário de edição de um canal.....	46
Figura 4-29: Página de visualização completa de um canal.....	47
Figura 4-30: Página de visualização completa de um canal do tipo - Imagem -.....	47

Figura 4-31: Confirmação de exclusão de um canal.....	48
Figura 4-32: Download de arquivo, no formato CSV, de um canal.....	48
Figura 4-33: Formulário de escolha dos parâmetros adicionais.....	49
Figura 4-34: Gráfico de um canal.....	49
Figura 4-35: Formulário de escolha dos parâmetros adicionais.....	50
Figura 4-36: Gráfico comparativo entre dois canais.....	50
Figura 4-37: Lista de medidas.....	51
Figura 4-38: Formulário de edição de uma medida.....	52
Figura 4-39: Confirmação de exclusão de uma medida.....	52
Figura 5-40: Visualização do equipamento - VandC - instalado no laboratório.....	54
Figura 5-41: Gráfico de variação de temperatura para dados reais.....	55
Figura 5-42: Gráfico de variação de voltagem para dados reais.....	55
Figura 5-43: Visualização do canal - Imagens -.....	56
Figura 5-44: Imagem obtida pelo equipamento - VandC -.....	56

Lista de Tabelas

Tabela 4-1: Campos do cabeçalho do arquivo de dados.....	28
Tabela 4-2: Campos de registros do arquivo de dados.....	28
Tabela 4-3: Linhas de um arquivo de registros de imagens.....	30
Tabela 4-4: Linhas de um arquivo de registros de erros.....	30
Tabela 4-5: Linhas de um arquivo de registros de medidas.....	31
Tabela 4-6: Linhas de um arquivo de solicitação de leitura de canais de medida.....	32
Tabela 4-7: Arquivo CSV criado para um equipamento.....	44
Tabela 4-8: Arquivo CSV criado para um canal.....	49
Tabela 5-9: Dados obtidos do equipamento - VandC -.....	54

Capítulo 1

1.1 - Introdução

1.2 - Tema

Um Sistema de Informação (SI) combina recursos humanos e computacionais para coletar, processar, armazenar, e transmitir dados com o objetivo de tornar o fluxo de informações mais confiável e menos burocrático para o usuário . As principais vantagens de um SI são: a otimização do fluxo de informação, permitindo maior agilidade e organização; a redução de custos e ganho de produtividade; a maior integridade e veracidade da informação; e a maior segurança de acesso à informação.

Sistemas baseados em tecnologia *web* têm a vantagem de serem acessíveis para qualquer usuário autorizado. O único requisito é que se tenha um navegador de Internet padrão, que por sua vez, se tornou um aplicativo presente em qualquer sistema operacional.

Este projeto refere-se ao desenvolvimento de um SI para auxiliar o gerenciamento de dados referentes aos protótipos de *hardware* desenvolvidos no Laboratório de Instrumentação e Fotônica (LIF) , pertencente ao Programa de Pós-Graduação e Pesquisa de Engenharia (COPPE) e à Escola Politécnica, ambos da UFRJ.

1.3 - Justificativa

O LIF é um laboratório multidisciplinar que atua em projetos de pesquisa e desenvolvimento nas áreas de instrumentação e fotônica, e consultoria na área de instrumentação . Por realizar parceria com empresas como Petrobrás, Eletronorte e Furnas, o LIF possui muitos dados de projetos que necessitam ser gerenciados.

O gerenciamento de dados é indispensável às empresas e também aos laboratórios de pesquisa, pois é através deste que toda análise e acompanhamento de projetos são realizados. Sistemas que permitem este gerenciamento proporcionam um

amplo controle sobre os protótipos de *hardware* desenvolvidos e suas medições para que novas tecnologias possam ser testadas com mais precisão e eficiência nos laboratórios, garantindo assim a qualidade dos projetos.

Atualmente, verifica-se a falta de um procedimento padrão de monitoramento dos dados de projetos que utilizam este tipo de telemetria.

1.4 - Objetivos

O presente projeto tem como objetivo o desenvolvimento de um sistema *web* para auxiliar o gerenciamento dos dados dos protótipos de *hardware* de campo do laboratório, bem como a aquisição dos parâmetros medidos pelos mesmos. Através deste projeto procura-se criar um procedimento padrão de aquisição, gestão e análise dos dados recebidos pelo LIF, provenientes de seus protótipos.

Para tal, é proposto um modelo de banco de dados capaz de identificar vários tipos de parâmetros como: voltagem, temperatura, potência, entre outros, e capaz de mapear todos os canais de medição dos protótipos. Então, deve ser desenvolvido um sistema para proporcionar a interação dos usuários com este banco de dados.

Adicionalmente, a variação de grandezas monitoradas ao longo do tempo deve ser analisada por meio de tabelas e gráficos. Estes dados podem ser exportados em arquivos no formato CSV (*Comma-Separated Values*) para análises mais detalhadas em editores de planilhas eletrônicas, tais como - *Excel* da Microsoft -.

Além disso, um protocolo de transferência de dados deve ser estipulado para que haja também uma padronização dos arquivos de dados enviados e recebidos pelo laboratório, tornando possível o correto tratamento dos dados.

A Figura 1 -1 mostra a comunicação entre o servidor do LIF, os protótipos de *hardware*, o banco de dados e a interface com os usuários.

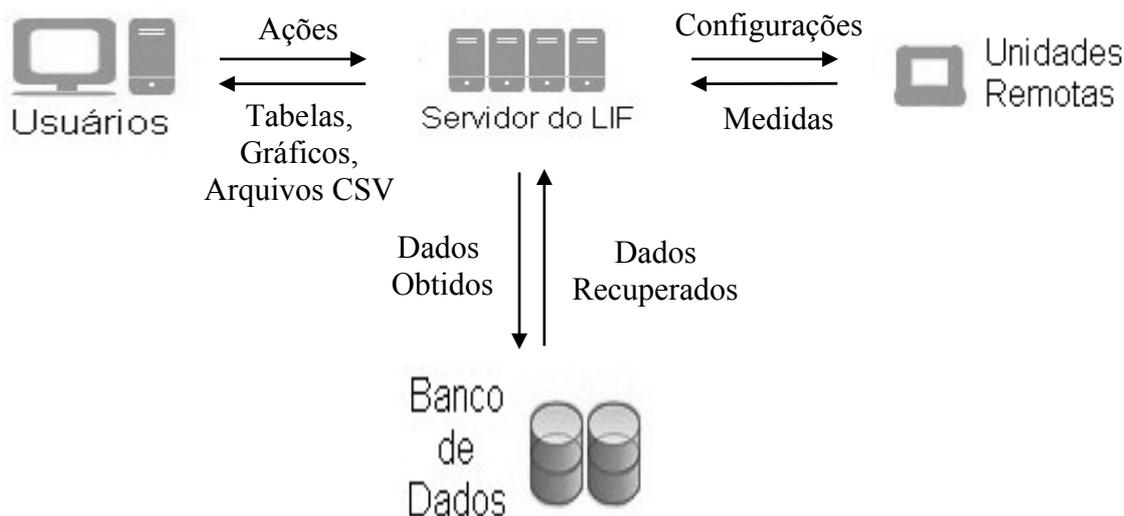


Figura 1-1: Interações entre os componentes do sistema.

1.5 - Metodologia

Seguindo os passos de um processo de desenvolvimento de *software*, primeiramente foi realizada a análise do problema para que pudesse ser feito o levantamento dos requisitos do sistema *web*.

Em seguida, foi elaborada uma modelagem para o banco de dados, genérica o suficiente para que atendesse a diferentes projetos do LIF, e uma especificação que apresentasse uma solução computacional para o gerenciamento do sistema. Foram definidas também as funcionalidades do sistema e as tecnologias utilizadas, de acordo com as necessidades encontradas na análise do problema e da disponibilidade do laboratório. Para estes primeiros passos houve a troca de conhecimentos e informações com os funcionários do laboratório, além de pesquisas na Internet.

A implementação do sistema *web* foi feita em blocos, tomando-se como base o padrão de arquitetura de *software* MVC (*Model-View-Controller*). Este tipo de desenvolvimento permite que cada parte do *software* seja implementada separadamente, para que assim possam ser feitos testes pontuais e, futuramente, sejam integradas umas às outras. Permite também que as tecnologias utilizadas em cada bloco sejam trocadas, caso necessário, sem que seja preciso modificar outros blocos do sistema.

Após o término do desenvolvimento do sistema, foram realizados testes em todos os seus blocos.

1.6 - Descrição

No Capítulo 2, Especificação do Projeto, será apresentada a análise do problema que motivou a elaboração do projeto, a análise dos requisitos do sistema e a solução proposta com a modelagem do banco de dados e a especificação de casos de uso do sistema.

Posteriormente, no Capítulo 3, Tecnologias Utilizadas, o padrão de arquitetura e as tecnologias escolhidas para o desenvolvimento do software são detalhados.

O Desenvolvimento do Sistema e o protocolo de transferência de dados são descritos no Capítulo 4.

No Capítulo 5, Resultados e Discussões, serão relatados os resultados obtidos pelos testes do sistema.

Por fim, o Capítulo 6 traz as conclusões do projeto.

Capítulo 2

1.7 - Especificação do Projeto

O sistema foi concebido para auxiliar o monitoramento e a análise de vários tipos de parâmetros referentes aos canais de medição dos protótipos de *hardware* de campo do LIF. Ao longo deste capítulo, serão mostradas as etapas iniciais do processo de desenvolvimento do *software*, que compreendem a análise do problema, o levantamento dos requisitos e, finalmente, a solução proposta para o projeto, com a modelagem do banco de dados e os casos de uso do sistema.

1.8 - Análise do Problema

O LIF desenvolve diversos projetos de Pesquisa e Desenvolvimento em parceria com empresas como: Petrobrás, Eletronorte e Furnas. Todos estes projetos possuem algum tipo de protótipo de *hardware* de campo que faz medições pertinentes ao objetivo para o qual foi projetado, assim como medições que permitem o controle de seu funcionamento, garantindo a qualidade na obtenção das medidas.

Alguns destes projetos não são de interesse para o presente trabalho, pois não possuem formas de comunicação de dados, ou possuem dados que necessitam apenas de monitoramento local, ou estão inseridos em redes privadas, que não permitem a comunicação externa de dados.

Há também alguns casos onde são implementados sistemas capazes de desempenhar as funções de telemetria, porém estas soluções estão relacionadas a problemas específicos de cada um destes projetos. Não há portanto, no LIF, uma padronização no que diz respeito a um sistema capaz de monitorar a aquisição de dados, armazenar estes valores em um banco de dados e fazer as análises necessárias para garantir o bom funcionamento dos protótipos desenvolvidos. A falta deste procedimento padrão de monitoramento dificulta a aquisição, a gestão e a realização das análises dos dados obtidos.

O desenvolvimento de um sistema capaz de realizar a coleta, processamento, armazenamento e análise de dados provenientes de um protótipo de *hardware* exige noções

avançadas de programação e banco de dados, bem como tempo para seu desenvolvimento, logo se faz necessária a implementação de um sistema padrão, que possa realizar estas funções para todos os projetos onde se aplica este tipo de análise.

1.9 - Análise dos Requisitos

O sistema foi desenvolvido para oferecer funcionalidades que resolvam os problemas citados no Item 2.1 de maneira padronizada. Para tal, foram elaborados requisitos para sua implementação, identificados a partir do estudo do ambiente e da interação com os funcionários do laboratório através de reuniões técnicas. Para atender às necessidades do LIF este sistema deve:

- Permitir ao usuário a inserção, a edição, a visualização detalhada e a exclusão de projetos, tipos de parâmetros, protótipos de *hardware*, canais de medida, medidas e imagens;
- Permitir que o usuário conecte o sistema ao protótipo de hardware, via FTP (File Transfer Protocol) , para aquisição de dados atualizados e também para o envio de arquivos de configuração. Para esta comunicação, deve ser elaborado um protocolo de transferência de dados;
- Permitir a verificação de falhas, tanto na comunicação com o equipamento, quanto na obtenção de dados pelo mesmo, através de seus canais de medição;
- Armazenar todos os dados obtidos dos equipamentos em um banco de dados, genérico o suficiente para atender aos variados projetos do laboratório, com suas características específicas;
- Possibilitar o monitoramento dos parâmetros de cada canal de medição ao longo do tempo, através de gráficos, e permitir a comparação das medições de dois canais, em certo período de tempo, por intermédio de gráficos comparativos;

- Disponibilizar as informações, criando arquivos no formato CSV, para análises mais detalhadas em editores de planilhas eletrônicas;
- Ser acessível a todos os integrantes do LIF, independentemente de sua localização geográfica, utilizando um navegador de Internet padrão.

O sistema deve ainda atender a outros requisitos como usabilidade, integridade dos dados e facilidade de manutenção.

1.10 - Sistema Proposto

Nesta fase, foi elaborada a especificação que apresenta uma solução computacional para o gerenciamento do sistema, o modelo do banco de dados e a definição de suas funcionalidades, de acordo com as necessidades encontradas nos itens 2.1 e 2.2, e da disponibilidade do laboratório.

1.10.1 - Modelagem de dados

Na grande maioria dos projetos a utilização de um banco de dados relacional é fundamental para o armazenamento e boa estruturação dos dados. Neste modelo de banco de dados, as tabelas associam-se obedecendo a regras de relacionamentos que consistem em associar um ou vários campos de uma tabela a um ou vários campos de outra tabela .

A modelagem de dados pode ser considerada uma das principais etapas do projeto, visto que é através dela que os campos de cada tabela referente aos dados de projetos, protótipos, tipos de parâmetros, canais e medidas, e o relacionamento entre as tabelas pode ser definido. Este modelo pode ser entendido como um mapeamento de informações sobre objetos do mundo real e suas relações com os outros objetos.

Esta modelagem deve ser feita visando evitar redundâncias, que são responsáveis por grandes perdas de desempenho, especialmente em sistemas de grande porte.

O modelo de banco de dados utilizado pode ser visualizado na Figura 2 -2. Este foi feito da maneira mais genérica possível, possibilitando a identificação de vários parâmetros relacionados aos mais variados projetos desenvolvidos no laboratório.

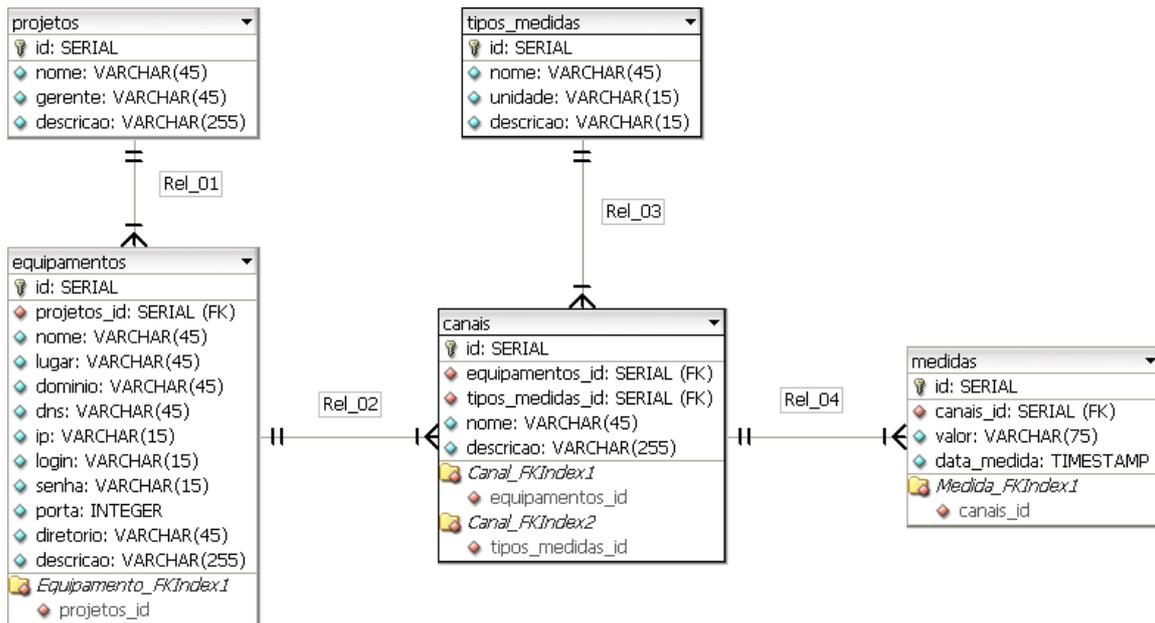


Figura 2-2: Modelo de banco de dados para o projeto.

Uma breve análise do modelo da Figura 2 -2 mostra a divisão em cinco entidades ou tabelas. O relacionamento entre elas, como dito anteriormente, depende da associação de um campo de uma tabela a um campo de outra tabela.

Cada projeto possui um identificador único - id -, e pode estar relacionado a diversos equipamentos, ou protótipos de *hardware*. No sentido oposto, observamos que cada equipamento só pode estar relacionado a um projeto. Esta relação de dependência é determinada pela inclusão do id do projeto no campo - projetos_id - do equipamento. Da mesma forma, verificamos que, pelo modelo proposto, cada canal é associado a um tipo de medição e a um equipamento, enquanto cada equipamento e cada tipo de medição podem estar relacionados a vários canais diferentes. Por fim, verificamos que cada canal pode possuir várias entradas na tabela de medidas, ao passo que cada medida está associada a apenas um canal.

É possível perceber que para cada tabela do banco há um campo, denominado id, que contém um identificador único para aquela tabela. A definição dos demais campos de cada tabela foi feita de acordo com as necessidades inicialmente projetadas para o sistema e foi modificada à medida que o sistema foi desenvolvido, adequando-os aos novos requisitos do projeto. Um exemplo desta modificação do banco de dados foi o caso da necessidade da inclusão de campos referentes ao servidor FTP na tabela de equipamentos.

1.10.2 - Casos de Uso

Os casos de uso são os modos através dos quais o sistema interage com o usuário ou com outros sistemas, a partir de determinadas ações deste usuário. É importante notar que estes casos de uso não descrevem como o *software* deverá ser desenvolvido, mas sim como deverá ser seu comportamento depois de pronto .

A documentação das funcionalidades do sistema facilita a realização dos testes do *software* e é uma ótima ferramenta para a manutenção do código-fonte. A Figura 2 -3 apresenta todas as possíveis interações do usuário com o sistema. O esquema mostra que o usuário pode listar os dados de todas as entidades do banco de dados, além de adicionar registros a elas, editá-los, fazer a visualização detalhada de cada um ou apagá-los do sistema.

O usuário pode também: fazer o *download* de arquivos no formato CSV contendo os registros de um canal ou protótipo selecionado; solicitar e obter leituras dos protótipos; e gerar gráficos com os valores das medidas de um canal. Os casos de uso são explicitados individualmente a seguir, visando à melhor compreensão de cada um.

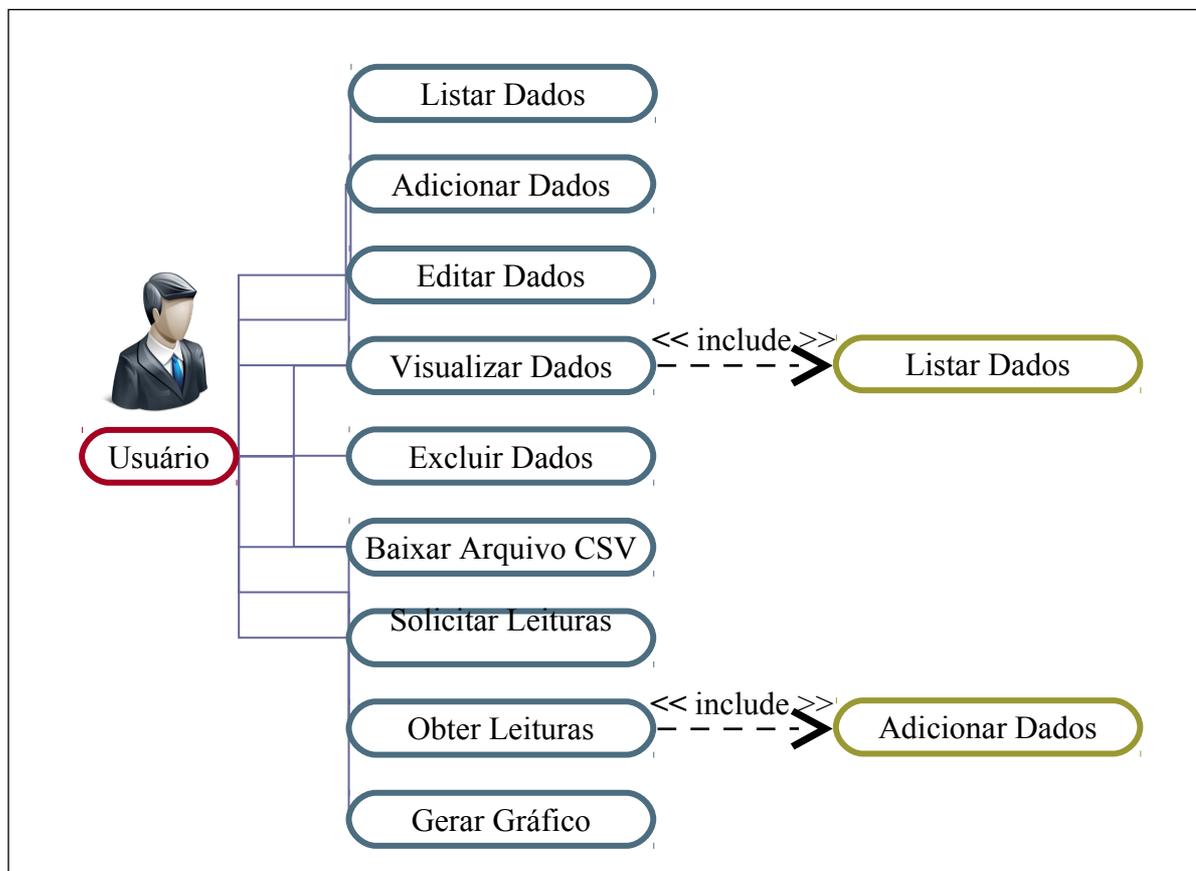


Figura 2-3: Diagrama de casos de uso.

1.10.2.a - Listar Dados

Este caso de uso é, na verdade, dividido em cinco diferentes casos, um para cada tabela do banco de dados. Cada um destes permite ao usuário visualizar uma tabela contendo todos os registros da entidade do modelo a que se referencia. Dependendo desta entidade, seus principais campos são listados.

São dadas também as opções de edição e exclusão de cada registro das tabelas. Além disso, todos os campos de nome das tabelas contém *links* para a visualização detalhada das entidades a que se referem.

1.10.2.b - Adicionar Dados

Assim como no caso - Listar Dados -, este é dividido em seis casos diferentes, um para cada tabela do banco de dados, além de um a mais para o caso particular em que o tipo da medição é uma imagem. Cada um destes casos permite ao usuário incluir um registro em uma das entidades do banco de dados. Dependendo desta entidade, um formulário diferente é elaborado.

O formulário criado para a adição de um novo protótipo de *hardware* contém também os campos relacionados à conexão do sistema com o servidor FTP passivo, onde serão encontrados os valores das medidas obtidas por todos os canais que estarão ligados a este protótipo.

As opções de inclusão manual de medidas e imagens foram implementadas, apesar destas, na prática, serem realizadas pelos canais dos protótipos.

Foram definidas também duas particularidades quanto ao tipo de parâmetro. O tipo - Imagem - guarda o endereço do arquivo de imagem, armazenado em uma pasta específica no servidor, em seu campo - Valor -, enquanto que o tipo - Erro - não pode ser utilizado para medições. Este tipo é utilizado para identificar erros no processo de medição dos canais, como será explicado no protocolo de transferência de dados, no Item 4.1.

1.10.2.c - Editar Dados

Este caso segue os mesmos moldes do caso - Adicionar Dados -.

1.10.2.d - Visualizar Dados

Novamente, este caso é dividido em cinco casos diferentes, um para cada tabela do banco de dados. Cada um destes permite ao usuário visualizar todos os campos de um registro de uma das entidades do banco de dados, assim como uma lista dos registros que estão ligados a este. Dependendo da entidade do registro selecionado, uma visualização diferente será apresentada.

Para o caso da visualização de um projeto, por exemplo, para cada registro da tabela de projetos são mostrados todos os seus campos, além de uma lista, conforme o caso de uso - Listar Dados -, dos equipamentos associados a este projeto.

1.10.2.e - Excluir Dados

Este caso permite que o usuário apague do banco de dados um registro de qualquer uma das entidades. Além disso, serão apagados também os registros de todas as entidades ligadas a esta. Esta exclusão exige uma confirmação do usuário, antes da sua execução, evitando assim a exclusão permanente de registros de forma acidental.

1.10.2.f - Baixar Arquivo CSV

A opção de *download* de arquivos no formato CSV é proporcionada na visualização de um canal e de um equipamento. No primeiro caso é disponibilizado um arquivo com todas as medições relacionadas àquele canal. Já no segundo caso é disponibilizado um arquivo com todas as medições dos canais relacionados àquele equipamento.

1.10.2.g - Solicitar Leituras

A opção de Solicitar Leituras é proporcionada na visualização de protótipos de *hardware*, já que estes possuem um servidor FTP, ou têm acesso a um, podendo então receber o arquivo de solicitação de leitura imediata de seus canais de medida.

1.10.2.h - Obter Leituras

Esta opção também é proporcionada na visualização de um protótipo, pelos mesmos motivos apresentados no caso de uso da opção Solicitar Leituras.

Além de serem feitas as leituras de todos os arquivos no diretório especificado no banco de dados para este protótipo em seu servidor FTP, é feita também a leitura das imagens, caso haja algum canal deste tipo, e são relatados os erros obtidos tanto na hora de medição dos parâmetros, quanto na hora de leitura dos arquivos. Estes arquivos e as mensagens de erro, como já explicitado anteriormente, são definidos no protocolo de transferência de dados, no Item 4.1.

1.10.2.i - Gerar Gráficos

Este caso é dividido em dois outros mais específicos. Ambos são habilitados na visualização de um canal de medida e possibilitam a escolha do intervalo de tempo mostrado no gráfico e do número de casas decimais dos valores medidos (mostrados em notação científica). No primeiro há a criação do gráfico do canal que está sendo visualizado, já no segundo, há a criação de um gráfico que faz a comparação temporal entre este canal e um segundo canal, escolhido junto com as outras opções do gráfico.

Capítulo 3

1.11 - Tecnologias Utilizadas

O sistema desenvolvido utiliza uma série de tecnologias de *software* para realizar as funções descritas no Item 2.3.2. Entre os aspectos considerados na escolha das tecnologias estão o desempenho e a flexibilidade oferecidos para determinada tarefa. Essas tecnologias são descritas a seguir.

1.12 - Banco de Dados (PostgreSQL)

Para o projeto em questão, foi utilizado o PostgreSQL , que atualmente é um dos SGBDs (Sistema Gerenciador de Bancos de Dados) de código aberto mais avançados. O PostgreSQL utiliza a linguagem SQL (*Structured Query Language*) para acessar os dados do banco.

A escolha do gerenciador PostgreSQL foi guiada pelas vantagens que oferece em relação a outros gerenciadores de código aberto. Um modo eficaz de verificar a eficiência de um gerenciador de dados é realizar um teste de propriedades ACID que engloba verificações de:

- **Atomicidade:** a transação deve ter todas as suas operações executadas, em caso de sucesso, ou nenhum resultado, de alguma operação, refletido sobre a base de dados, em caso de falha. Ou seja, após o término de uma transação (*commit* ou *abort*), a base de dados não deve refletir resultados parciais da transação;
- **Consistência:** regras de integridade dos dados são asseguradas, ou seja, as transações não podem quebrar as regras do banco de dados;

- Isolamento: o resultado de uma transação executada concorrentemente a outra deve ser o mesmo que o de sua execução de forma isolada. Operações exteriores a uma dada transação jamais verão esta transação em estados intermediários;
- Durabilidade / Estabilidade: os efeitos de uma transação, em caso de sucesso, (*commit*) são permanentes mesmo depois de um término anormal.

O PostgreSQL possui todas estas propriedades. Além disso, o sistema de segurança do PostgreSQL permite utilizar mecanismos de autenticação como senhas e acesso controlado com base no endereço de IP.

O *software* utilizado como interface com o banco de dados foi o pgAdmin III, que é distribuído junto com o PostgreSQL. O *software* de modelagem utilizado foi o DB Designer Fork, pois através desse é possível gerar os *scripts* de criação do banco, após a criação do modelo ilustrado pela Figura 2 -2, sem escrever uma linha sequer de *script* SQL.

1.13 - Servidor Web (Apache HTTP Server)

O Apache HTTP Server é um servidor *web* desenvolvido pela Apache Software Foundation que se caracteriza por ser extremamente configurável, robusto e de alto desempenho. O Apache é um *software* livre, o que significa que qualquer um pode estudar ou alterar seu código-fonte, além de poder utilizá-lo gratuitamente. Graças ao trabalho, muitas vezes voluntário, de vários desenvolvedores de *software*, o Apache continua sendo o servidor *web* mais usado no mundo .

1.14 - Padrão de Arquitetura de Software MVC (*Model-View-Controller*)

O fluxo de informação entre o banco de dados e a interface com o usuário é fundamental para qualquer sistema. É natural, então pensar em vincular estas duas partes para reduzir a quantidade de codificação e melhorar o desempenho do *software*. Esta abordagem, no entanto, implica vários problemas significativos, como o fato de que a interface com o usuário tende a mudar com muito mais frequência do que a base de dados. Outro problema encontrado diz respeito ao fato de sistemas incorporarem lógicas de negócio que vão muito além da simples transmissão de dados.

Desta maneira, há necessidade de se modularizar o desenvolvimento, de forma a facilitar as modificações em uma parte do sistema, sem que esta implique em mudanças nas outras partes. Para solucionar estes problemas, foi feita a implementação do sistema tomando-se por base o padrão de arquitetura de *software* MVC, que possibilita esta separação entre lógica de negócio e interface com o usuário. Esse tipo de desenvolvimento permite que cada parte do *software* seja implementada separadamente, para que assim possam ser feitos testes pontuais e futuramente sejam integradas umas às outras.

O padrão MVC separa o modelo de dados, a apresentação destes dados e as ações do usuário em três componentes básicos, listados a seguir.

1.14.1 - Camada de modelo (*Model*):

Representa os dados da aplicação e as regras do negócio que governam o acesso e a modificação dos dados. Deve conter todas as informações importantes e que podem ser editadas pelo usuário.

O modelo responde aos pedidos de informação sobre seu estado atual (enviados pela camada de interface), e responde às instruções de mudança de seu estado (enviadas pela camada de controle). Esta camada deve ser totalmente desacoplada das demais, ou seja, não toma conhecimento das outras.

1.14.2 - Camada de controle (*Controller*):

É a camada que define o comportamento da aplicação, pois processa e responde a eventos, geralmente ações do usuário, e pode invocar alterações no modelo. Em uma aplicação *web* estas ações do usuário correspondem a cliques de botões ou seleções de menus. Há, normalmente, um controlador para cada conjunto de funcionalidades relacionadas, no caso deste projeto existe um controlador para cada tabela do banco de dados.

Com base na ação do usuário e no resultado do processamento do modelo, o controlador seleciona uma página a ser exibida como parte da resposta da solicitação do usuário. Para isto, é necessário que a camada de controle conheça as outras camadas.

1.14.3 - Camada de interface (*View*):

É a camada que apresenta o conteúdo de uma parte particular do modelo e encaminha para o controlador as solicitações de ação do usuário. Esta também acessa os dados do modelo através do controlador e define como estes dados devem ser apresentados. Assim como o modelo, esta camada não deve conhecer nada sobre as demais.

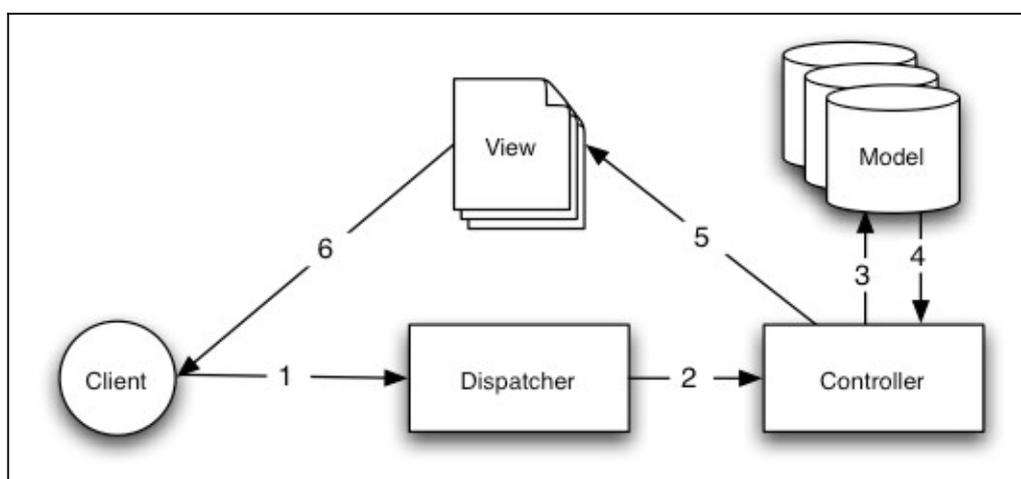


Figura 3-4: Requisição básica de MVC.
Fonte: CakePHP .

Como pode ser visto pela Figura 3 -4, uma requisição básica de MVC segue a ordem:

- O usuário interage com a interface de alguma forma (por exemplo, o usuário aperta um botão);
- A camada de controle manipula o evento da interface do usuário;
- A camada de controle acessa a camada de modelo, possivelmente atualizando-a de uma maneira apropriada, baseada na interação com o usuário (por exemplo, atualizando os dados de cadastro do usuário);

- A camada de interface utiliza a camada de modelo para gerar uma interface apropriada (por exemplo, mostrando na tela os dados que foram alterados juntamente com uma confirmação).

É importante ressaltar que tanto a camada de interface quanto a camada de controle, dependem da camada de modelo. Entretanto, esta última não depende de nenhuma das outras duas. Este é um dos principais benefícios desta separação em camadas, permitir que o modelo seja construído e testado independentemente do resto do sistema, da forma com que ocorrerá a interação com o usuário.

As vantagens na utilização deste padrão de arquitetura são:

- Suporte às múltiplas interfaces com o usuário. A interface com o usuário pode possuir várias páginas, no caso de uma aplicação *web*, mostrando diferentes dados ao mesmo tempo;
- Acomodação às mudanças de *layout* da camada de interface. A cor, estilo de fonte, ou tamanho da exibição, não fazem diferença para as outras camadas do sistema, podendo haver uma personalização da interface;
- Possibilidade de desenvolvimento em paralelo. Isto se deve ao fato das camadas serem isoladas.

Enquanto que as desvantagens são:

- Pequeno aumento da complexidade da solução. Este padrão de projeto introduz novas camadas e níveis de abstração, aumentando ligeiramente a complexidade da solução;
- Aumento do tempo necessário para modelagem das aplicações. O custo pode não compensar para emprego em aplicações pequenas.

1.15 - Linguagem de Programação PHP (*PHP: Hypertext Preprocessor*)

A maior parte do sistema será implementada empregando-se programas escritos com a versão 5 da linguagem de programação PHP . Esta é uma linguagem de programação de *script*, pois é executada do interior de documentos HTML (*Hypertext Markup Language*), muito utilizada na criação de aplicações *web*. Apesar de ser uma linguagem de fácil aprendizagem e de utilização para programas pequenos, dinâmicos e simples, PHP é uma linguagem que suporta a modelagem orientada a objetos.

A escolha desta linguagem deveu-se ao prévio conhecimento adquirido pelo LIF nesta plataforma, além de diversas vantagens oferecidas, como velocidade de processamento, versatilidade, e eficiência. É a linguagem *web* mais popular no mercado, possuindo ampla documentação disponível, o que a torna uma linguagem bastante robusta, uma vez que está sempre passando por testes e recebendo atualizações e correções de possíveis erros reportados pelos desenvolvedores.

1.16 - Framework CakePHP

Um *framework*, na área de desenvolvimento de *software*, consiste em um *software* utilizado para implementar a estrutura padrão de uma aplicação. Um *framework* tem o objetivo de facilitar o desenvolvimento, através de reutilização de código, permitindo que os programadores passem mais tempo preocupados com os requisitos do *software*, ao invés de ficar lidando com detalhes do funcionamento do sistema, de nível mais baixo. Sendo assim, os *frameworks* facilitam o trabalho com tecnologias complexas e às vezes até nos obriga a implementar códigos mais consistentes, com menos erros.

CakePHP é um *framework* escrito em PHP que tem como principal objetivo oferecer uma estrutura que possibilite aos programadores de PHP desenvolverem rapidamente aplicações sem perder flexibilidade. Este *framework* baseia-se no padrão MVC descrito no Item 3.3.

Suas funcionalidades, utilizadas no desenvolvimento do sistema, incluem métodos de integração com o banco de dados, métodos de auxílio à implementação do modelo de execução AJAX, que será descrito a seguir, métodos de paginação dos dados obtidos do banco de dados na forma de tabelas, estruturação do projeto em diretórios, facilitando sua compreensão, dentre muitas outras funcionalidades oferecidas.

1.17 - Modelo de execução AJAX (*Asynchronous Javascript And XML*)

O modelo de execução AJAX refere-se à união de várias tecnologias, que são capazes de tornar o navegador mais interativo, promovendo a troca de informações entre o servidor e o usuário de forma assíncrona, ou seja, processando mais de uma opção por vez.

Em um modelo clássico de aplicação *web* as ações do usuário na interface disparam uma solicitação para o servidor, que processa o pedido recuperando dados, e então retorna uma página HTML para o cliente, ou seja, enquanto o servidor está executando alguma tarefa, o cliente espera. Esse tempo de espera é indesejado quando se pretende usar a *web* como base para aplicativos.

Uma aplicação baseada em AJAX provê uma camada intermediária para controlar essa comunicação entre o usuário e o servidor. O mais importante desse processo é que os recursos usados do servidor são bem menores que o modelo tradicional, pois a única requisição feita é de conteúdo e não da página como um todo, que já está carregada no cliente e pronta para receber os dados. Qualquer resposta que não dependa do servidor, como por exemplo validação de dados, a camada intermediária a executa de maneira autônoma. Todo processo de exibição final fica por conta da máquina do cliente, fazendo assim com que o servidor ganhe em desempenho. O esquema do modelo AJAX é apresentado na Figura 3 -5.

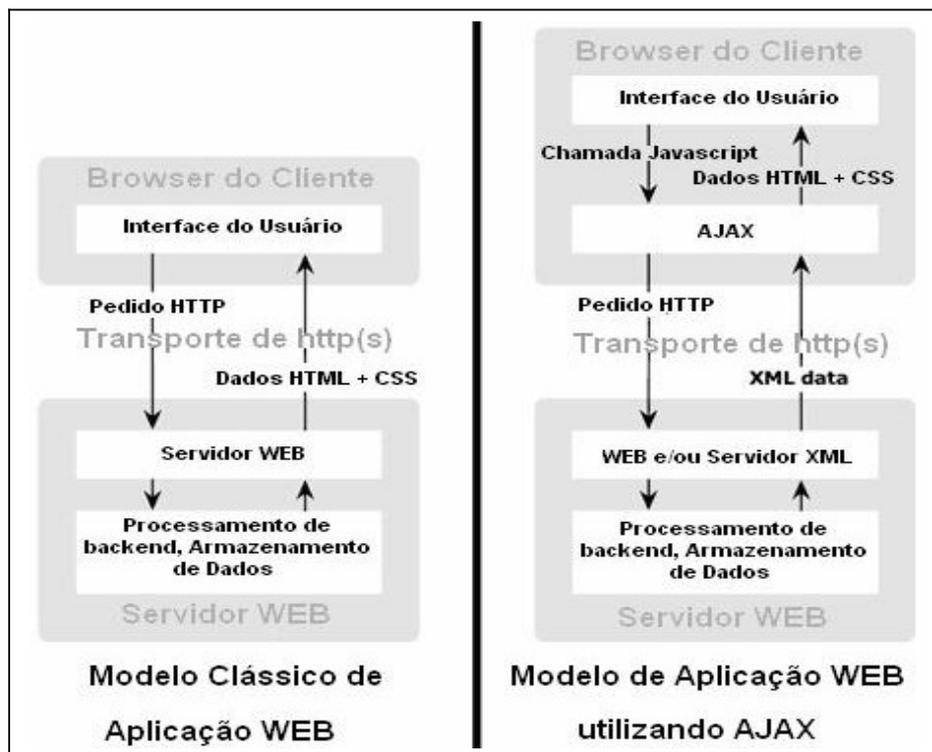


Figura 3-5: Comparação do modelo tradicional de execução com o modelo AJAX.
 Fonte: “Ajax: A New Approach to Web Applications” .

Cada uma das tecnologias utilizadas no modelo AJAX possui sua forma de exibição e interação com o usuário. O AJAX incorpora as tecnologias listadas a seguir.

1.17.1 - Folhas de Estilo CSS (*Cascading Style Sheets*)

Com a utilização de folhas CSS é possível mudar o estilo de marcações HTML em todos os documentos que fazem referência às mesmas, poupando assim o trabalho de modificar diversos documentos. Seu principal benefício é prover a separação entre o formato e o conteúdo de um documento.

1.17.2 - XML (*Extensible Markup Language*)

XML é uma linguagem de marcação de texto, capaz de descrever diversos tipos de dados. Seu propósito principal é a facilidade de compartilhamento de informações através da Internet. O XML não é uma simples linguagem de marcação pré-definida, mas sim uma metalinguagem, uma linguagem usada para descrever outras linguagens, que permite que seu usuário defina a sua própria marcação.

Enquanto o HTML dá ênfase à apresentação de dados, o XML permite descrever os dados, concentrando-se na estrutura destes. É utilizada na troca de dados, já que permite sua formatação em um padrão que pode ser entendido por qualquer sistema.

1.17.3 - JavaScript

JavaScript é a linguagem de *script* que une todas as outras tecnologias listadas. Nesta linguagem de programação a carga de processamento é realizada no lado do usuário, pelos navegadores, que são os encarregados por interpretar as instruções JavaScript e executá-las. Com esta linguagem pode-se criar efeitos especiais nas páginas e definir a interatividade com o usuário. JavaScript tem sintaxe semelhante à linguagem Java.

1.17.4 - DOM (*Document Object Model*)

DOM é uma especificação do *World Wide Web Consortium (W3C)*, independente de linguagem, onde se pode dinamicamente alterar e editar a estrutura, conteúdo e estilo de um documento HTML ou XML

DOM define uma maneira padrão de acesso e manipulação de documentos HTML ou XML, apresentando seus elementos em uma árvore estruturada. De acordo com este padrão, cada elemento de um documento é tratado como um nó da árvore, podendo conter atributos, outros elementos ou textos, que são tratados como as folhas da árvore. Isto permite que o documento seja processado, por uma linguagem de programação, como JavaScript, e os resultados desse processamento sejam incorporados de volta ao próprio documento.

A API (*Application Programming Interface*) DOM oferece uma maneira padrão de se acessar os elementos de um documento, além de se poder trabalhar com cada um desses elementos separadamente, e por esses motivos criar páginas bastante dinâmicas.

A Figura 3 -6 mostra a visão de um documento HTML em estrutura de árvore.

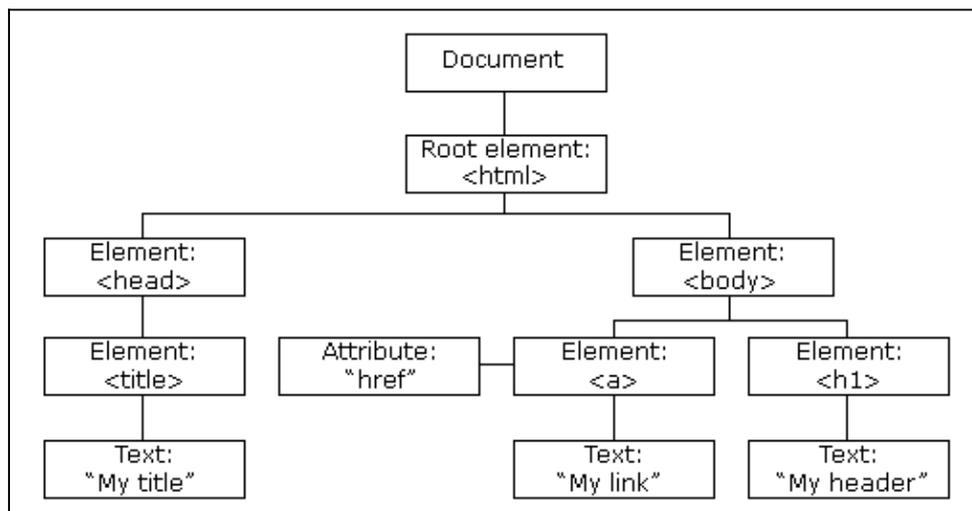


Figura 3-6: Documento HTML em estrutura de árvore.
Fonte: W3C .

1.17.5 - XMLHttpRequest

XMLHttpRequest é um objeto JavaScript que torna possível a comunicação assíncrona com o servidor, sem a necessidade de recarregar a página por completo. Este objeto faz parte da especificação do DOM.

O modelo de execução AJAX foi utilizado para garantir maior interatividade nos sistemas desenvolvidos, bem como aumento da velocidade das respostas ao cliente. Neste projeto, não foi empregada a linguagem XML para estruturar os dados enviados pelos controladores para as páginas de visualização, mas sim funções já implementadas no *framework* CakePHP.

1.18 - CSV (Comma-Separated Values)

CSV é um [formato de arquivo](#) que armazena dados tabelados. Por serem bastante simples, os arquivos deste formato são comuns em todas as plataformas e suportados por qualquer editor de planilha eletrônica.

O CSV é uma implementação particular de [arquivos de texto](#) separados por um delimitador, que emprega a [vírgula](#) (,) e a [quebra de linha](#) de texto (\n) para separar os valores. O formato CSV também usa as [aspas](#) (") em campos nos quais são empregados os [caracteres](#) reservados (vírgula e quebra de linha de texto).

Todos os arquivos do protocolo de transferência de dados, descrito no Item 4.1, e também o arquivo disponível para *download*, segundo o caso de uso - Baixar Arquivo CSV -, utilizam este formato, com o ponto e vírgula (;) como separador de campos.

Capítulo 4

1.19 - Desenvolvimento do Sistema

Ao longo deste capítulo, será detalhado o desenvolvimento do sistema, feito a partir das especificações apresentadas no Capítulo 2 e das tecnologias listadas no Capítulo 3. Primeiramente, será definido o protocolo elaborado para permitir a comunicação bilateral entre o servidor e as unidades remotas, que são os protótipos de campo. Posteriormente, será explicitada a implementação do *software* em camadas. Por fim, serão apresentadas as principais páginas de interface do usuário com o sistema.

1.20 -Protocolo de Transferência de Dados

O protocolo de transferência foi estipulado para padronizar a troca de informação entre o LIF e as unidades remotas (UR). A Figura 4 -7 demonstra a comunicação bilateral entre o sistema *web*, instalado no servidor do LIF, e uma de suas unidades remotas, realizada por solicitação do usuário.

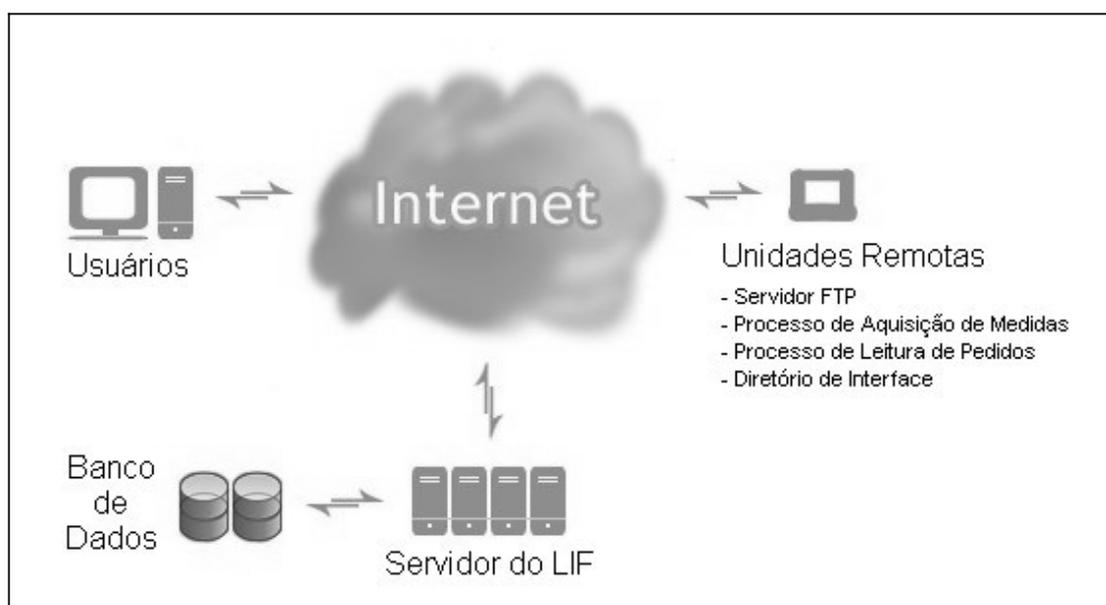


Figura 4-7: Conexão entre o servidor do LIF, o usuário do sistema e a unidade remota.

Na Figura 4 -7, é mostrado que a unidade remota executa um processo, em intervalos de tempo regulares, sendo responsável pela geração de arquivos que contêm os valores obtidos das medições feitas por seus canais, colocando-os no diretório de interface. Outro processo é executado continuamente, para verificar a existência de um arquivo, no diretório de interface, e solicitar o início imediato do processo de aquisição de medidas. A unidade remota possui também um servidor FTP, que permite que o sistema *web* tenha acesso ao diretório de interface da unidade remota. Este servidor opera em modo passivo, ou seja, ao se conectar ao servidor, o cliente recebe o número da porta através da qual deve realizar a transferência de dados. O sistema *web* projetado, quando recebe um pedido do usuário para obtenção de medidas, conecta-se ao equipamento e faz a leitura de todos os arquivos contendo medições realizadas por seus canais. Quando o usuário do sistema *web* solicita a aquisição imediata de medidas, é enviado ao equipamento um arquivo com esta solicitação. As definições do protocolo de transferência de dados são apresentadas nos próximos itens.

O diagrama da Figura 4 -8 demonstra a troca arquivos entre o servidor do laboratório e uma de suas unidades remotas.

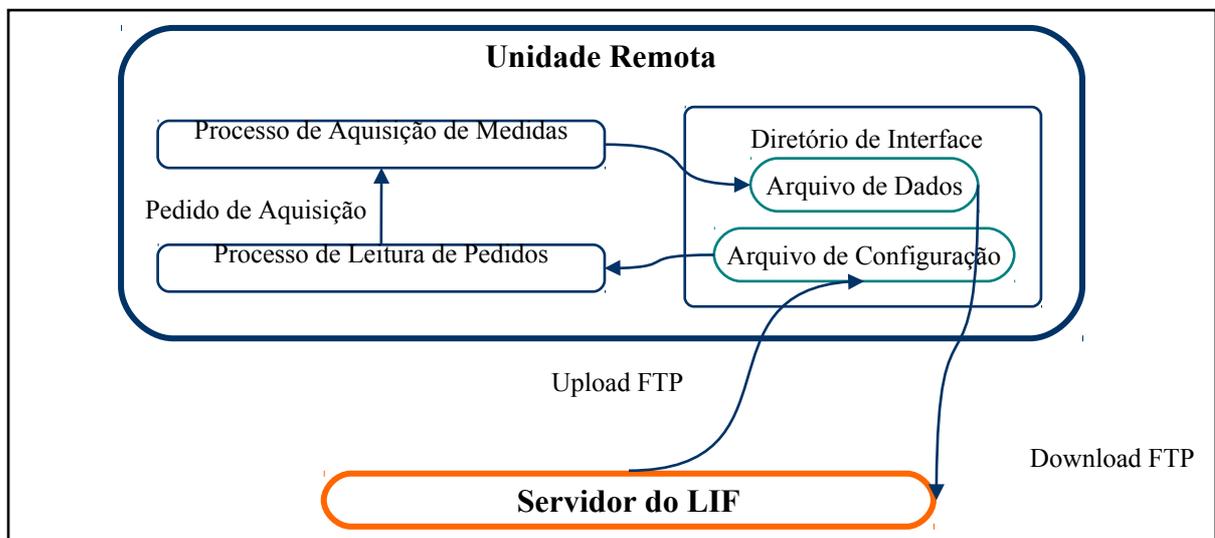


Figura 4-8: Troca de arquivos entre o servidor e uma unidade remota.

1.20.1 - Diretório de Interface Cliente/Equipamento

O diretório de interface cliente/equipamento tem como função armazenar os arquivos gerados pelos equipamentos e pelo cliente. As regras descritas a seguir devem ser seguidas para evitar a concorrência no acesso aos arquivos.

1.20.1.a - Recepção de dados pelo cliente

Os equipamentos coletam dados dos canais e criam arquivos - LIF-xxxxx.DAT -, onde xxxxx é um grupo de caracteres alfanuméricos gerados automaticamente. Para gerar um arquivo de dados, os equipamentos obedecem à seguinte ordem:

- Criar o arquivo - LIF-xxxxx - no diretório de interface sem a extensão - DAT -. Esse arquivo contém um ou mais blocos de informação dos canais de medida;
- Renomear o arquivo para - LIF-xxxxx.DAT - após ter finalizado a gravação dos dados.

O seguinte procedimento deve ser executado pelo cliente a fim de obter o arquivo:

- Verificar a existência de um arquivo - LIF-xxxxx.DAT -;
- Transferir o arquivo de dados;
- Remover o arquivo de dados do equipamento;
- Repetir o procedimento até que todos os arquivos - LIF-xxxxx.DAT - tenham sido processados. Os arquivos sem a extensão - DAT - devem ser desconsiderados.

1.20.1.b - Transmissão de dados pelo cliente

Os equipamentos aceitam arquivos do cliente contendo comandos de leitura de dados de um ou mais canais. Para enviar os comandos, o cliente deve colocá-los num arquivo de nome - WEB-xxxxx.DAT -, onde xxxxx é um grupo de caracteres alfanuméricos gerados automaticamente. Para enviar um arquivo de comandos, o cliente deve obedecer aos seguintes procedimentos:

- Criar o arquivo - WEB-xxxxx -;
- Transferir o arquivo - WEB-xxxxx - para o equipamento via FTP;

- Renomear o arquivo para - WEB-xxxxx.DAT -.

O seguinte procedimento deve ser executado pelos equipamentos para ler o arquivo enviado pelo cliente:

- Verificar a existência de um arquivo - WEB-xxxxx.DAT -;
- Ler o arquivo e processar as mensagens contidas no mesmo;
- Remover o arquivo;
- Repetir o procedimento até que todos os arquivos - WEB-xxxx.DAT - tenham sido processados. Os arquivos sem a extensão - DAT - devem ser desconsiderados.

1.20.2 - Formato do Arquivo de Dados

O CSV é um formato de arquivos que armazena dados tabelados, conforme descrito no Item 3.7. Os arquivos de dados utilizam este formato, contendo ao menos um bloco de informação, sendo composto das seguintes áreas:

- Uma estrutura - *File Header Frame* - no início do arquivo;
- Uma ou mais estruturas - *Data Frame* -.

1.20.2.a - Cabeçalho do Arquivo (*File Header Frame*)

O cabeçalho do arquivo descreve as características de criação do arquivo em campos delimitados pelo caractere ponto e vírgula (;). Sua principal finalidade é identificar o equipamento a que pertencem os canais lidos. O seu formato é o seguinte:

<code><Filename>;<Timestamp>;<EquipId></code>

Onde:

Campo	Descrição
<Filename>	Nome do arquivo.
<Timestamp>	Data e hora de criação do arquivo (YYYY-MM-DD HH:mm:SS).
<EquipId>	Identificador do equipamento.

Tabela 4-1: Campos do cabeçalho do arquivo de dados.

1.20.2.b - Registro de Dados (File Data Frame)

Após o cabeçalho do arquivo deverá haver uma ou mais linhas de dados, denominadas - *File Data Frame* -, que serão registros de dados cujos campos são separados pelo caractere ponto e vírgula (;). O formato de cada registro de dados é o seguinte:

<TipId>;<CanalId>;<Timestamp>;<Valor>

Onde:

Campo	Descrição
<TipId>	Identificador do tipo de medida obtida.
<CanalId>	Identificador do canal de medida.
<Timestamp>	Data e hora de obtenção da medida (YYYY-MM-DD HH:mm:SS).
<Valor>	Campo de dados dependente do tipo de medida. Seu conteúdo será descrito no Item 4.1.3.

Tabela 4-2: Campos de registros do arquivo de dados.

1.20.3 - Tipos de *Frames* de Dados

Existem três formatos de *frames* de dados específicos para os arquivos - LIF-xxxxx.DAT - e um formato para os arquivos - WEB-xxxxx.DAT -. Cada *frame* de dados possui um - Valor - diferente, descrito a seguir.

1.20.3.a - Registro de Leitura de Dados

Tipo Imagem

Quando o canal de leitura de dados obtém imagens, elas são armazenadas no diretório de interface com o cliente. Este canal é associado a um tipo, denominado - Imagem -, cujo

identificador é obtido da tabela de tipos de medida do banco de dados. O campo - Valor - de um registro de leitura deste tipo contém o endereço da imagem, armazenada no diretório de interface, que deve ser obtida pelo cliente.

Tipo Erro

Este tipo de registro é utilizado para informar erros ocorridos no processo de leitura do equipamento. O campo - TipoId -, neste caso, não está associado ao canal, mas sim ao tipo denominado - Erro -, cujo identificador é obtido da tabela de tipos de medida do banco de dados. O campo - Valor - de um registro de leitura deste tipo contém a descrição do erro ocorrido.

Outros Tipos

Para todos os outros tipos de parâmetros medidos, o campo - Valor - contém o valor obtido pelo canal de medida.

1.20.3.b - Solicitação de Ciclo de Leitura

Este *frame* contém uma requisição de leitura dos canais de um protótipo de *hardware* e, portanto, deve ser gerado pelo cliente. Os campos - Timestamp e Valor - do frame devem permanecer vazios. Quando o equipamento processar este comando, deverá disparar a leitura dos dados dos canais e, uma vez finalizada, o equipamento deverá gerar o arquivo - LIF-xxxxx.DAT - no diretório de interface.

1.20.4 - Exemplos

Neste capítulo serão apresentados alguns exemplos de arquivos, com a finalidade de ilustrar o formato dos registros e dos arquivos de dados.

1.20.4.a - Frames de Leitura de Dados

Tipo Imagem

Supondo que o canal 15 é do tipo - Imagem -, identificado pelo - id - 1, o equipamento 12 cria o arquivo LIF-zJkLX.DAT no diretório de interface com o seguinte conteúdo:

```
LIF-zJkLX.DAT;2009-06-20 12:00:00;12  
1;15;2009-06-20 12:00:35;imagem1.jpg
```

Este arquivo contém as seguintes informações:

Linha	Descrição
<i>LIF-zJkLX.DAT;2009-06-20 12:00:00;12</i>	Cabeçalho do arquivo.
<i>1;15;2009-06-20 12:00:35;imagem1.jpg</i>	Registro de imagem (imagem1.jpg) no diretório de interface.

Tabela 4-3: Linhas de um arquivo de registros de imagens.

Tipo Erro

Supondo que tenha sido solicitada ao equipamento 7 a leitura do canal 15, associado ao tipo 1, e o processo não tenha sido concluído com êxito, o equipamento irá reportar a ocorrência do erro através do tipo - Erro -, de - id - 6, conforme exemplo a seguir:

```
LIF-8q6Gy.DAT;2009-06-20 12:00:00;7  
6;15;2009-06-20 12:00:38;"Falha na resposta do canal, canal não encontrado."
```

Este arquivo contém as seguintes informações:

Linha	Descrição
<i>LIF-8q6Gy.DAT;2009-06-20 12:00:00;7</i>	Cabeçalho do arquivo.
<i>6;15;2009-06-20 12:00:38;"Falha na resposta do canal, canal não encontrado."</i>	Registro de erro de leitura do canal 15.

Tabela 4-4: Linhas de um arquivo de registros de erros.

Outros Tipos

Supondo que o tipo - Temperatura - é identificado pelo - id - 10 e que o canal 14 é deste tipo, o equipamento 12 cria o arquivo LIF-zJkLX.DAT no diretório de interface com o seguinte conteúdo:

```
LIF-zJkLX.DAT;2009-06-20 12:00:00;12
10;14;2009-06-20 12:00:35;2.31e+1
```

Este arquivo contém as seguintes informações:

Linha	Descrição
<i>LIF-zJkLX.DAT;2009-06-20 12:05:00;12</i>	Cabeçalho do arquivo.
<i>10;14;2009-06-20 12:00:35;2.31e+1</i>	Registro de temperatura (23.1 graus Celsius) em notação científica.

Tabela 4-5: Linhas de um arquivo de registros de medidas.

Os exemplos anteriores são básicos, contendo apenas um registro. Em situações reais de operação, um arquivo de dados gerado por um equipamento pode ter registros de leitura de qualquer tipo. No exemplo a seguir, ocorreu um erro na leitura do canal 2 do equipamento 3, enquanto que os canais 1 e 3 foram lidos com sucesso. O canal 1 é associado ao tipo - Corrente -, de - id - 12, o canal 3 ao tipo - Imagem -, de - id - 9, enquanto que o canal 2 está, neste caso, associado ao tipo - Erro -, de - id - 6.

```
LIF-8q6G3y.DAT;2009-06-20 12:00:00;3
12;1;2009-06-20 12:00:10;2.31e-3
6;2;2009-06-20 12:00:10;Falha na conexão com o canal.
9;3;2009-06-20 12:00:10;imagem1.png
12;1;2009-06-20 12:05:10;1.38e-3
6;2;2009-06-20 12:05:10;Falha na conexão com o canal.
9;3;2009-06-20 12:05:10;imagem2.png
```

1.20.4.b - Frames de Solicitação de Leitura

No diretório de interface o cliente cria o arquivo WEB-aVbGfc.DAT com uma solicitação de leitura dos canais 1 e 2 do equipamento 4, associados aos tipos 7 e 3 respectivamente, com o seguinte conteúdo:

```
WEB-aVbGfc.DAT;2009-06-20 12:00:00;4
7;1
3;2
```

Este arquivo contém as seguintes informações:

Linha	Descrição
<i>WEB-aVbGfc.DAT;2009-06-20 12:00:00;4</i>	Cabeçalho do arquivo.
<i>7;1</i>	Solicitação de leitura do canal 1.
<i>3;2</i>	Solicitação de leitura do canal 2.

Tabela 4-6: Linhas de um arquivo de solicitação de leitura de canais de medida.

Como resposta à execução deste comando, o equipamento irá gerar um arquivo - LIF-xxxxxx.DAT - contendo frames de leitura dos dados solicitados. Caso algum comando não seja executado, uma das linhas do arquivo indicará o erro encontrado.

1.21 - Desenvolvimento

1.21.1 - Introdução

A implementação do sistema foi feita utilizando programação orientada a objetos (POO), que foi criada para tentar aproximar o mundo real do mundo virtual. Na POO, implementa-se um conjunto de classes que definem os objetos presentes no sistema de *software*. Cada classe determina o comportamento (definido nos métodos) e estados possíveis (atributos) de seus objetos, assim como o relacionamento com outros objetos.

1.21.2 - Organização das Classes

O desenvolvimento do sistema foi realizado tomando-se como base o padrão de arquitetura de *software* MVC. Este padrão de arquitetura é implementado pelo *framework* CakePHP, que provê uma estrutura básica de organização, desde nomes de arquivos até nomes de tabelas de bancos de dados, mantendo toda a aplicação consistente e lógica .

A separação do projeto em camadas de: modelo de dados, funcionalidades e visualização de dados, é ilustrada, para a tabela de equipamentos do banco de dados, na , e explicada nos próximos itens.

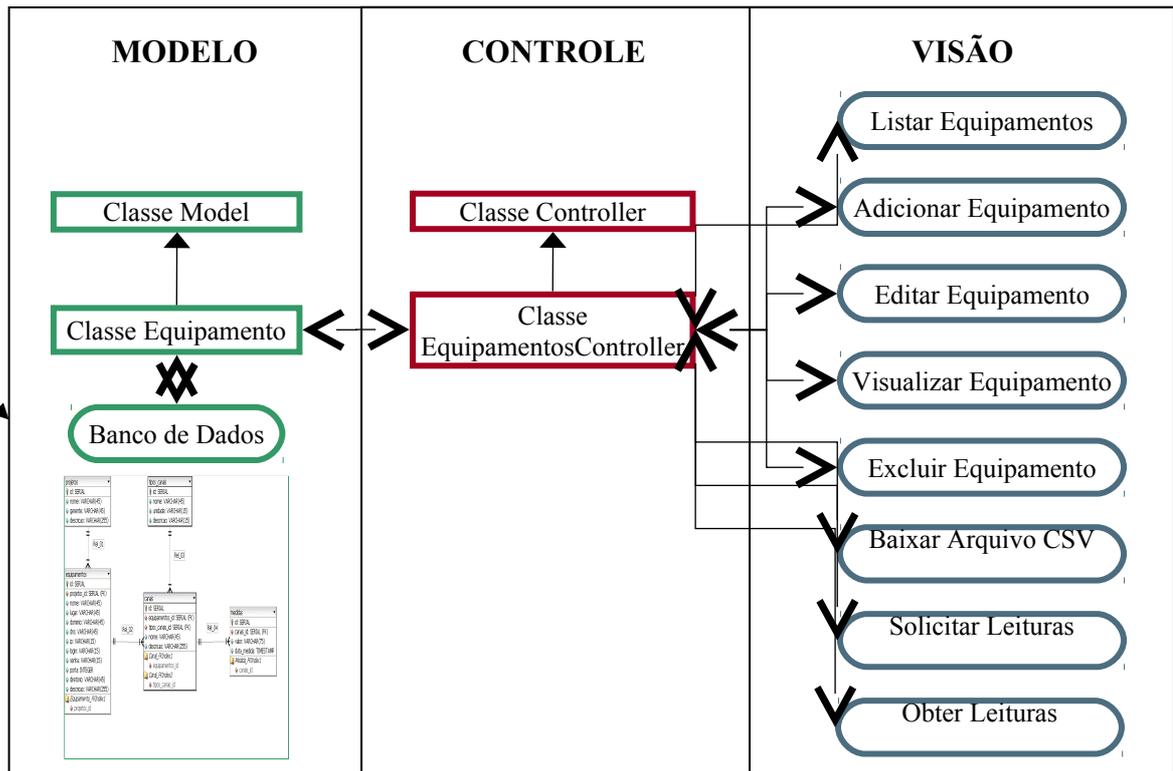


Figura 4-9: Modelo MVC para a tabela de equipamentos, com suas funcionalidades.

Camada de Modelo (*Model*):

O primeiro passo na implementação do sistema foi criar, para cada tabela do banco de dados, uma classe com o nome da tabela no singular, seguindo convenção do CakePHP. O seguimento destas convenções garante, não apenas um melhor entendimento da estrutura do projeto, mas também promove a conexão entre as classes implementadas, que é feita de forma automática pelo *framework*. Estas classes são derivadas da classe base - *Model* -, implementada pelo CakePHP, que possui atributos e métodos de conexão com o banco de dados e de manipulação de seus dados de forma eficiente.

Estas classes criadas para as tabelas do banco de dados possuem métodos e atributos que fazem a validação dos dados que serão inseridos no banco, além de atributos que as relacionam com as outras classes da camada de modelo, conforme definido pela modelagem do banco. Possuem também, métodos para manipulação dos arquivos CSV criados para cada canal de medição e protótipo de *hardware*, e métodos para manipulação dos arquivos de gráfico criados.

Camada de Controle (*Controller*):

Após a implementação das classes do modelo, foram criadas as classes de controle do sistema, que definem o comportamento geral do *software*. Para cada tabela do banco de dados foi desenvolvida uma classe, com o nome da tabela seguida da expressão - *Controller* -, como pode ser visto na Figura 4 -9, também seguindo convenção do *framework*. Estas classes são derivadas da classe base - *Controller* -, implementada pelo CakePHP, que possui métodos e atributos que promovem a interação do modelo com as páginas de visualização.

Estas classes de controle possuem métodos que processam e respondem a todas as ações do usuário, selecionando a página a ser exibida, com base no resultado do processamento do modelo.

Camada de Interface (*View*):

A camada de visão do sistema é composta por páginas HTML, que utilizam o modelo AJAX. Para cada método criado na camada de controle, foi criada uma página na camada de interface com o usuário, como mostrado na Figura 4 -9.

Estas páginas encaminham para o controlador as solicitações de ação do usuário. Estas páginas também acessam os dados do modelo através dos controladores e definem como estes dados devem ser apresentados.

O diagrama de seqüência para o acesso ao banco de dados é ilustrado pela Figura 4 -10, facilitando o entendimento do relacionamento entre as classes e páginas criadas.

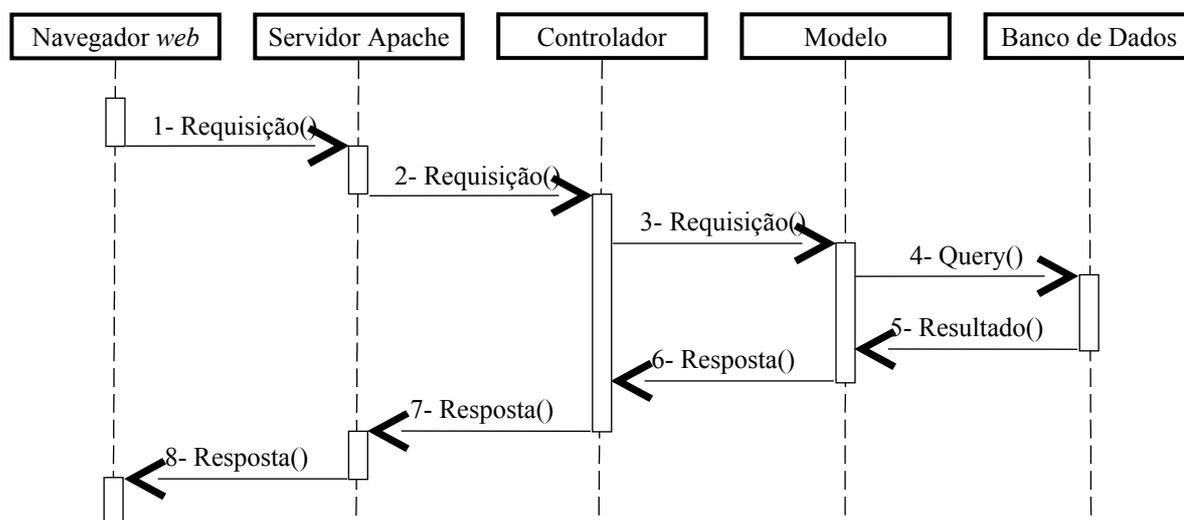


Figura 4-10: Diagrama de seqüência para acesso ao banco de dados.

Como observado pela Figura 4 -10, o usuário, pelo seu navegador *web*, faz uma requisição de acesso ao banco de dados ao servidor Apache para, por exemplo, editar um equipamento. Este envia tal tarefa para o controlador, no caso, - EquipamentosController -, que através do método - *Edit* -, identifica a ação de acesso ao banco de dados e invoca a classe Equipamento, da camada de modelo. Esta classe realiza a validação dos dados, e então, através de métodos da classe - *Model* -, acessa o banco e realiza as modificações feitas pelo usuário em algum registro da tabela de equipamentos. Após a consulta ou inserção no banco de dados, este retorna ao cliente o resultado de uma busca ou a confirmação de uma operação. Esta resposta percorrerá todo o diagrama de volta e será exibida ao usuário.

1.22 - Páginas de Interface com o Usuário

As páginas de interface promovem a interação do usuário com as funcionalidades do sistema, listadas no Item 2.3.2. A página inicial do sistema é mostrada na Figura 4 -11. Nesta página pode ser visto o menu principal do sistema, circundado de vermelho, que contém as opções - Listar - e - Adicionar - para cada tabela do banco de dados. Este menu principal pode ser visto em qualquer uma das páginas do sistema. Na Figura 4 -11 o sub-menu de - Medidas -, circundado de verde, mostra estas opções.



Figura 4-11: Página inicial do sistema.

1.22.1 - Projetos

As funcionalidades do sistema relacionadas aos projetos são listadas a seguir. Como o menu principal e o rodapé das páginas seguem o padrão mostrado na Figura 4 -12, este serão omitidos das demais figuras, assim como o navegador *web*.

1.22.1.a - Listar Projetos

Como observado na Figura 4 -12, o nome, identificador e gerente de cada projeto são listados, além de serem dadas as opções de edição e exclusão de cada um. Os nomes e identificadores de cada projeto possuem *links* para sua visualização detalhada.



Id	Nome	Gerente	Ações
1	Projeto Teste 1		Editar Excluir
2	Projeto Teste 2	Gerente 2	Editar Excluir
3	Projeto Teste 3	Gerente	Editar Excluir

Figura 4-12: Lista de projetos do banco de dados.

1.22.1.b - Adicionar/Editar Projeto

As opções de inclusão e edição de projeto são idênticas, como descritas nos itens 2.3.2.b e 2.3.2.c, e permitem que cada campo de um registro da tabela de projetos seja alterado. Na opção de edição os dados atuais do projeto são carregados nos campos, como mostra a Figura 4 -13.

Editar Projeto

Nome
Projeto Teste 1

Gerente
[Campo vazio]

Descricao
[Campo de texto grande]

Enviar

Figura 4-13: Formulário de edição de um projeto.

1.22.1.c - Visualizar Projeto

Na Figura 4 -14 é possível ver todos os campos associados a um determinado projeto, assim como uma lista dos equipamentos associados a este. São dadas também as opções de inclusão de um equipamento, edição e exclusão do projeto.

Projeto Adicionar Equipamento Editar Excluir

Id 2
Nome Projeto Teste 2
Gerente Gerente 2
Descrição -

Equipamentos

Página 1 de 1, mostrando 2 equipamentos de um total de 2, começando pelo 1, terminando no 2

Id	Nome	Lugar	Ações
4	Equipamento Teste 1	Laboratório	Editar Excluir
5	Equipamento Teste 2		Editar Excluir

<< | | >>

Figura 4-14: Página de visualização completa de um projeto.

1.22.1.d - Excluir Projeto

Esta opção é dada tanto na visualização de um projeto, quanto na listagem de todos os projetos. Ao clicar nesta opção, o usuário recebe uma mensagem de confirmação informando que a exclusão de um projeto implica na exclusão de todos os equipamentos, canais e medidas associados a este projeto. Esta mensagem *pop-up* pode ser vista na Figura 4 -15.

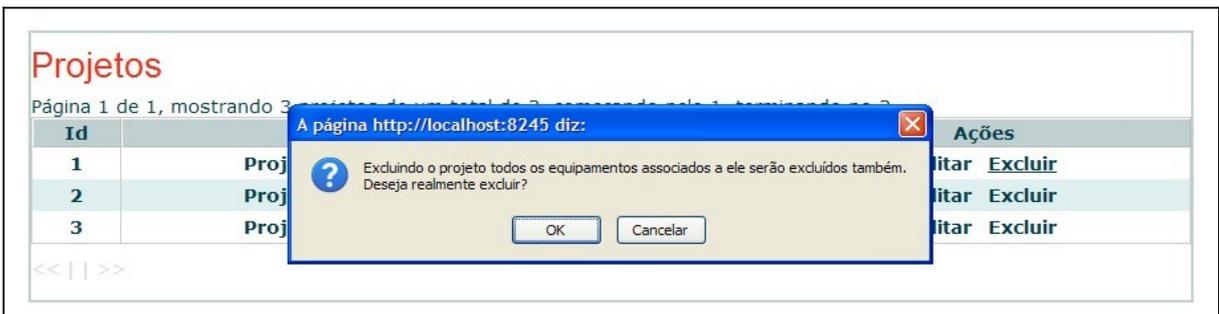


Figura 4-15: Confirmação de exclusão de um projeto.

1.22.2 - Tipos de Medida

As funcionalidades do sistema relacionadas aos tipos de parâmetros são listadas a seguir.

1.22.2.a - Listar Tipos

Pode-se observar pela Figura 4-16, que o nome, identificador e unidade de cada tipo de medição que pode ser efetuada pelos protótipos de campo são listados, além de serem dadas as opções de edição e exclusão de cada um. Os nomes e identificadores de cada um destes tipos possuem *links* para sua visualização detalhada.

Tipos de Medida			
Página 1 de 2, mostrando 10 tipos de um total de 11, começando pelo 1, terminando no 10			
Id	Nome	Unidade	Ações
9	Capacitância	C	Editar Excluir
6	Condutância	S	Editar Excluir
1	Corrente	A	Editar Excluir
7	Erro		Editar Excluir
2	Imagem		Editar Excluir
8	Indutância	H	Editar Excluir
5	Potência	W	Editar Excluir
10	Resistência		Editar Excluir
3	Temperatura	°C	Editar Excluir
4	Voltagem	V	Editar Excluir

Figura 4-16: Lista de tipos de medidas.

1.22.2.b - Adicionar/Editar Tipo

As opções de inclusão e edição de tipo são idênticas, e permitem que cada campo de um registro da tabela de tipos seja alterado. Na opção de edição os dados atuais do tipo são carregados nos campos, como mostra a Figura 4 -17.

O formulário, intitulado "Editar Tipo de Medida", contém os seguintes campos:

- Nome:** Campo de texto com o valor "Capacitância".
- Unidade:** Campo de texto com o valor "C".
- Descricao:** Campo de texto grande e vazio.
- Enviar:** Botão de submissão.

Figura 4-17: Formulário de edição de um tipo.

1.22.2.c - Visualizar Tipo

Na Figura 4 -18 é possível ver todos os campos associados a um determinado tipo, assim como uma lista dos canais associados a este. São dadas também as opções de edição e exclusão do tipo.

A página de visualização, intitulada "Tipo de Medida", apresenta os seguintes dados:

Tipo de Medida Editar Excluir

Id: 2
Nome: Imagem
Unidade: -
Descrição: -

Canais

Página 1 de 1, mostrando 2 canais de um total de 2, começando pelo 1, terminando no 2

Id	Equipamento	Nome	Ações
64	Equipamento de imagens	Imagens	Editar Excluir
67	Equipamento de imagens	Imagens2	Editar Excluir

<< | | >>

Figura 4-18: Página de visualização completa de um tipo.

1.22.2.d - Excluir Tipo

Esta opção é dada tanto na visualização de um tipo, quanto na listagem de todos os tipos. Ao clicar nesta opção, o usuário recebe uma mensagem de confirmação informando que a exclusão de um tipo implica na exclusão de todos os canais e medidas associados. Esta mensagem *pop-up* pode ser vista na Figura 4 -19.

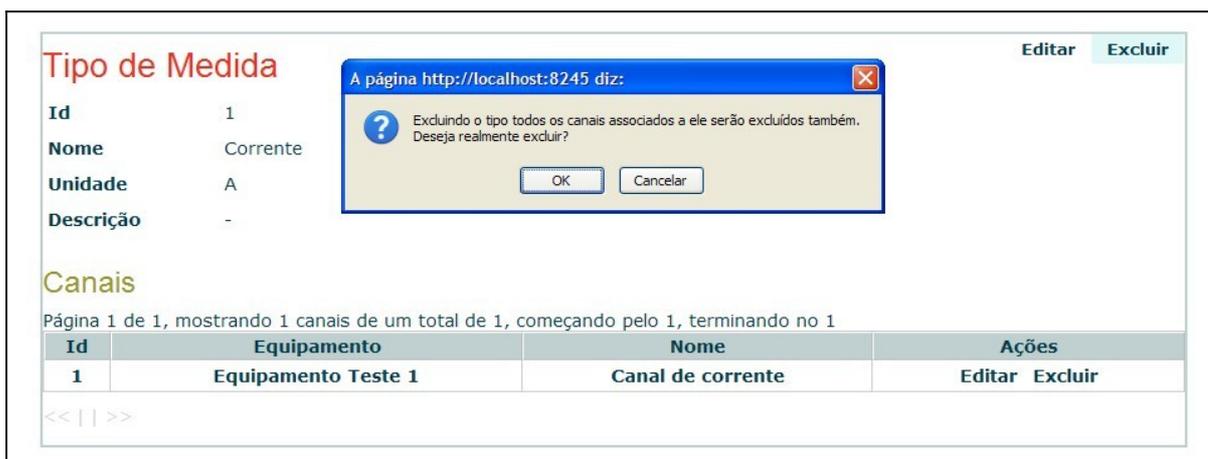


Figura 4-19: Confirmação de exclusão de um tipo.

1.22.3 - Equipamentos

As funcionalidades do sistema relacionadas aos equipamentos são listadas a seguir.

1.22.3.a - Listar Equipamentos

Pode-se observar pela Figura 4 -20, que o identificador, projeto, nome e local de instalação de cada equipamento são listados, além de serem dadas as opções de edição e exclusão de cada um. Os nomes e identificadores de cada equipamento possuem *links* para sua visualização detalhada, enquanto que o nome do projeto possui um *link* para sua visualização.

Equipamentos				
Página 1 de 1, mostrando 5 equipamentos de um total de 5, começando pelo 1, terminando no 5				
Id	Projeto	Nome	Lugar	Ações
4	Projeto Teste 2	Equip. 1	Laboratório	Editar Excluir
5	Projeto Teste 2	Equip. 2		Editar Excluir
1	Projeto Teste 1	Equipamento 1	Laboratório	Editar Excluir
2	Projeto Teste 1	Equipamento 2	Lugar	Editar Excluir
23	Projeto Teste 1	Equipamento de Imagens		Editar Excluir

<< | | >>

Figura 4-20: Lista de equipamentos.

1.22.3.b - Visualizar Equipamento

Na Figura 4-21 é possível ver todos os campos associados a um determinado equipamento, assim como uma lista dos canais associados a este. São dadas também as opções de: adição de um canal; edição e exclusão do equipamento; *download* de arquivo, no formato CSV, contendo as medidas de todos os canais associados a este equipamento; obtenção dos arquivos de medidas do equipamento; e envio de solicitação de leitura de medidas.

Equipamento		Baixar CSV Obter Leituras Solicitar Leitura Adicionar Canal Editar Excluir
Id	1	
Projeto	Projeto Teste 1	
Nome	Equipamento 1	
Lugar	Laboratório	
Domínio	lif.coppe.ufrj.br	
DNS	INFO14	
IP	-	
Login FTP	login	
Senha FTP	senha	
Porta FTP	-	
Diretório FTP	-	
Descrição	-	

Canais			
Página 1 de 1, mostrando 2 canais de um total de 2, começando pelo 1, terminando no 2			
Id	Tipo	Nome	Ações
13	Condutância	Canal de condutância	Editar Excluir
1	Corrente	Canal de corrente	Editar Excluir

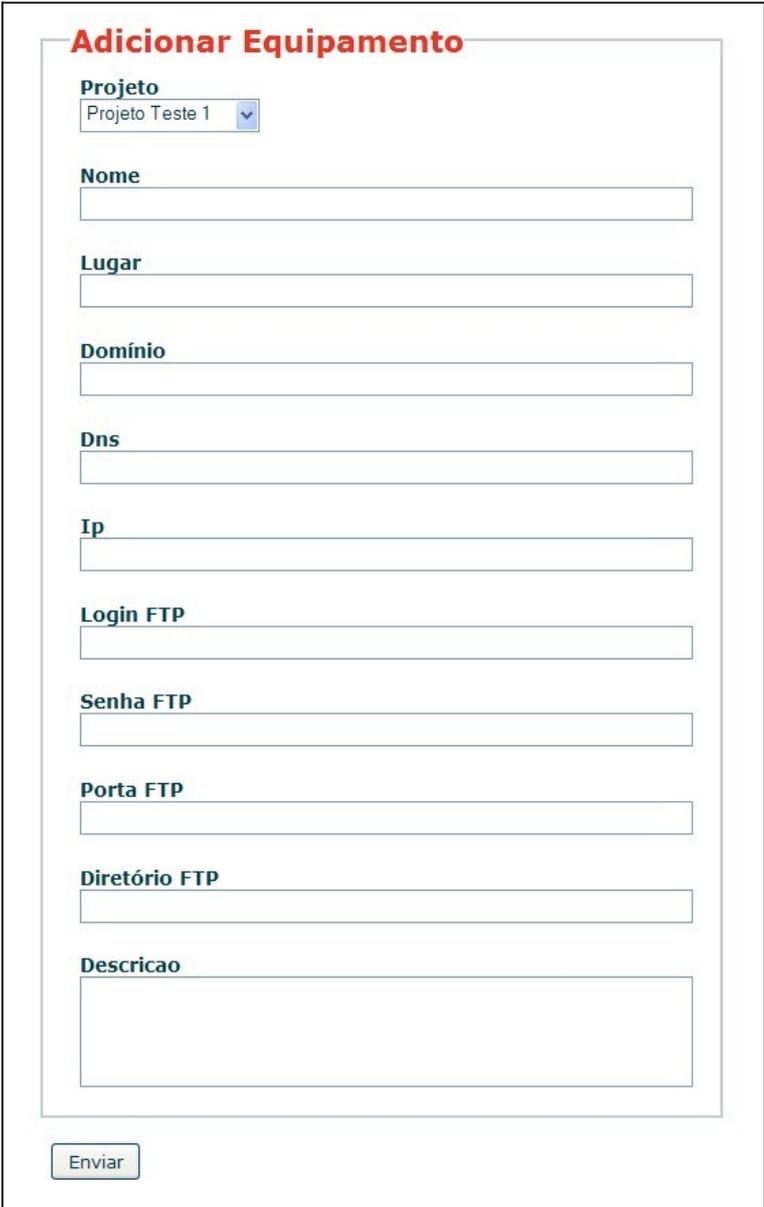
<< | | >>

Figura 4-21: Página de visualização completa de um equipamento.

1.22.3.c - Adicionar/Editar Equipamento

As opções de inclusão e edição de equipamentos são idênticas, e permitem que cada campo de um registro da tabela de equipamentos seja alterado. A Figura 4 -22 mostra a inclusão de um novo equipamento.

Como mostrado no Item 2.3.2.b, o formulário de criação de um equipamento contém também os campos relacionados à conexão do sistema com o seu servidor FTP passivo. Estes campos são: domínio, dns, ip, login FTP, senha FTP, porta FTP e diretório FTP.



Adicionar Equipamento

Projeto
Projeto Teste 1 ▾

Nome

Lugar

Domínio

Dns

Ip

Login FTP

Senha FTP

Porta FTP

Diretório FTP

Descricao

Figura 4-22: Formulário de criação de um equipamento.

1.22.3.d - Excluir Equipamento

Esta opção é dada na listagem de todos os equipamentos, na visualização de um equipamento e na visualização do projeto ao qual este equipamento está associado. Ao clicar nesta opção, o usuário recebe uma mensagem de confirmação informando que a exclusão de um equipamento implica na exclusão de todos os canais e medidas associados. Esta mensagem *pop-up* pode ser vista na Figura 4 -23.

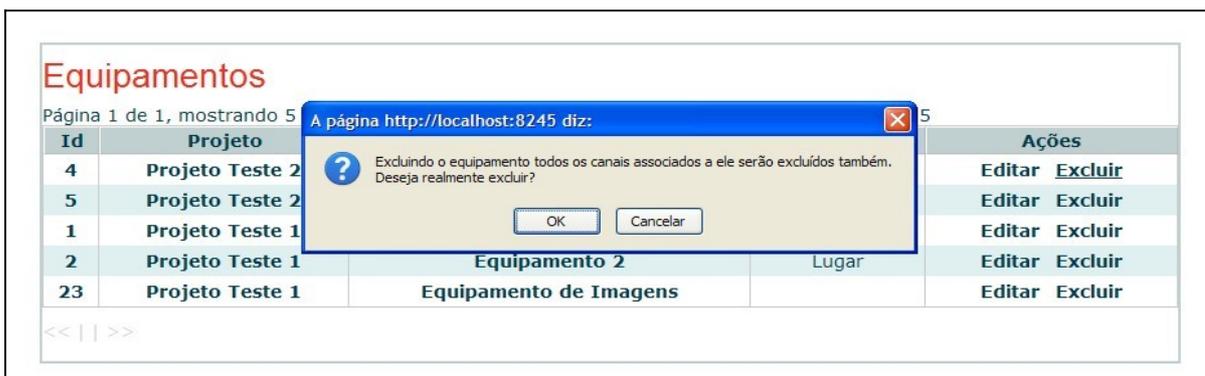


Figura 4-23: Confirmação de exclusão de um equipamento.

1.22.3.e - Baixar CSV

Na Figura 4 -24 é possível ver a janela de confirmação de *download* do arquivo que contém todos os dados dos canais de medida deste equipamento. O arquivo criado é do formato mostrado na Tabela 4 -7.



Figura 4-24: *Download* de arquivo, no formato CSV, de um equipamento.

Projeto	Projeto Teste 1 (id 1)
Equipamento	Equipamento 1 (id 1)
Canal	Canal de corrente (id 1)
Tipo	Corrente (id 1, unidade A)
Data medida	Valor
12/5/2009 21:25:03	2,00E-03
12/5/2009 21:26:04	3,00E-03
12/5/2009 21:26:23	3,46E-03
12/5/2009 21:30:03	2,50E-03
12/5/2009 21:30:04	2,60E-03
Canal	Canal de condutância (id 13)
Tipo	Condutância (id 6, unidade S)
Data medida	Valor
12/5/2009 21:25:03	1,00E+03
12/5/2009 21:26:04	2,00E+03
12/5/2009 21:26:23	3,40E+03
12/5/2009 21:30:03	2,65E+03
12/5/2009 21:30:04	2,90E+03

Tabela 4-7: Arquivo CSV criado para um equipamento.

1.22.3.f - Obter Leituras

Ao clicar nesta opção, o usuário recebe uma mensagem informando se foram obtidas medidas ou se ocorreu algum erro. Este erro pode estar relacionado à falha na conexão com a unidade remota, a falhas nas medições feitas pelos canais ou a falhas na leitura dos arquivos contendo as medições. Um exemplo de mensagem de erro na conexão com o equipamento pode ser visto na Figura 4 -25.



Figura 4-25: Mensagem de erro na conexão com o equipamento.

1.22.3.g - Solicitar Leitura

Ao clicar nesta opção, o usuário recebe uma mensagem informando se foi enviada a solicitação ou se ocorreu algum erro de conexão com a unidade remota. No exemplo da Figura 4 -26, uma mensagem de sucesso no envio de solicitação de leitura é mostrada.



Figura 4-26: Mensagem de sucesso no envio de solicitação de leitura.

1.22.4 - Canais

As funcionalidades do sistema relacionadas aos canais de medição são listadas a seguir.

1.22.4.a - Listar Canais

Pode-se observar pela Figura 4 -27, que o identificador, projeto, equipamento, tipo de medida e nome de cada canal são listados, além de serem dadas as opções de edição e exclusão de cada um. Os nomes e identificadores de cada canal possuem *links* para sua visualização detalhada, enquanto que os nomes dos projetos, equipamentos e tipos de medida possuem *links* para suas respectivas visualizações.

The screenshot shows a table titled "Canais" with the following data:

Canais					
Página 1 de 1, mostrando 5 canais de um total de 5, começando pelo 1, terminando no 5					
Id	Projeto	Equipamento	Tipo	Nome	Ações
13	Projeto Teste 1	Equipamento 1	Condutância	Canal de condutância	Editar Excluir
1	Projeto Teste 1	Equipamento 1	Corrente	Canal de corrente	Editar Excluir
2	Projeto Teste 1	Equipamento 2	Potência	Canal de potência	Editar Excluir
64	Projeto Teste 1	Equipamento de Imagens	Imagem	Imagens	Editar Excluir
67	Projeto Teste 1	Equipamento de Imagens	Imagem	Imagens2	Editar Excluir

Navigation controls: << | | >>

Figura 4-27: Lista de canais.

1.22.4.b - Adicionar/Editar Canal

As opções de inclusão e edição de canais são idênticas, e permitem que cada campo de um registro da tabela de canais seja alterado. A Figura 4 -28 mostra a edição de um canal. Neste formulário de edição, o campo - Equipamento - varia de acordo com o - Projeto - selecionado.



O formulário, intitulado "Editar Canal", contém os seguintes campos:

- Projeto:** Menu suspenso com "Projeto Teste 2" selecionado.
- Equipamento:** Menu suspenso com "Equip. 1" selecionado.
- Tipo:** Menu suspenso com "Condutância" selecionado.
- Nome:** Campo de texto contendo "Canal de condutância".
- Descricao:** Campo de texto grande e vazio.
- Enviar:** Botão de submissão.

Figura 4-28: Formulário de edição de um canal.

1.22.4.c - Visualizar Canal

Na Figura 4 -29 é possível ver todos os campos associados a um determinado canal, assim como uma lista das medidas associadas a este, com seus valores em notação científica. São dadas também as opções de: adição de uma medida; edição e exclusão do canal; *download* de arquivo, no formato CSV, contendo todas as medidas deste canal; criação de gráfico a partir das medidas do canal; e criação de gráfico comparativo entre o canal visualizado e outro canal, conforme descrito no Item 2.3.2.i. Pode-se notar na Figura 4 -29 também, a utilização de paginação das medidas relacionadas a este canal, circulada de vermelho.

Canal Baixar CSV Gráfico Comparativo Gráfico Adicionar Medida Editar Excluir

Id 1
Projeto Projeto Teste 1
Equipamento Equipamento 1
Tipo de Medida Corrente
Nome Canal de corrente
Descrição -

Medidas

Página 1 de 2, mostrando 10 canais de um total de 12, começando pelo 1, terminando no 10

Id	Data	Valor	Ações
2	12-05-2009 21:25:03	2.0000e+0	Editar Excluir
3	12-05-2009 21:26:04	3.0000e+0	Editar Excluir
19	12-05-2009 21:26:23	3.4570e+0	Editar Excluir
10	12-05-2009 21:30:03	2.5000e+0	Editar Excluir
11	12-05-2009 21:30:04	2.6000e+0	Editar Excluir
12	12-05-2009 21:30:05	4.3500e+0	Editar Excluir
13	12-05-2009 21:30:07	6.4000e+0	Editar Excluir
14	12-05-2009 21:30:08	7.4000e+0	Editar Excluir
15	12-05-2009 21:30:15	2.2000e+0	Editar Excluir
17	12-05-2009 21:30:19	5.6000e+0	Editar Excluir

<< | 1 | 2 | >>

Figura 4-29: Página de visualização completa de um canal.

Para o caso do canal ser do tipo - Imagem -, ao invés de ser mostrado o valor da medida, é dado um *link* para a visualização da imagem, como pode ser visto na Figura 4 -30.

Canal Baixar CSV Adicionar Medida Editar Excluir

Id 64
Projeto Projeto Teste 1
Equipamento Equipamento de Imagens
Tipo de Medida Imagem
Nome Imagens
Descrição -

Medidas

Página 1 de 1, mostrando 1 canais de um total de 1, começando pelo 1, terminando no 1

Id	Data	Imagem	Ações
43	12-07-2009 00:48:23		Editar Excluir

<< | >>

Figura 4-30: Página de visualização completa de um canal do tipo - Imagem -.

1.22.4.d - Excluir Canal

Esta opção é dada na listagem de todos os canais, na visualização de um canal e nas visualizações do equipamento e do tipo de medida, aos quais este canal está associado. Ao clicar nesta opção, o usuário recebe uma mensagem de confirmação informando que a exclusão de um canal implica na exclusão de todas as medidas associadas. Esta mensagem *pop-up* pode ser vista na Figura 4 -31.

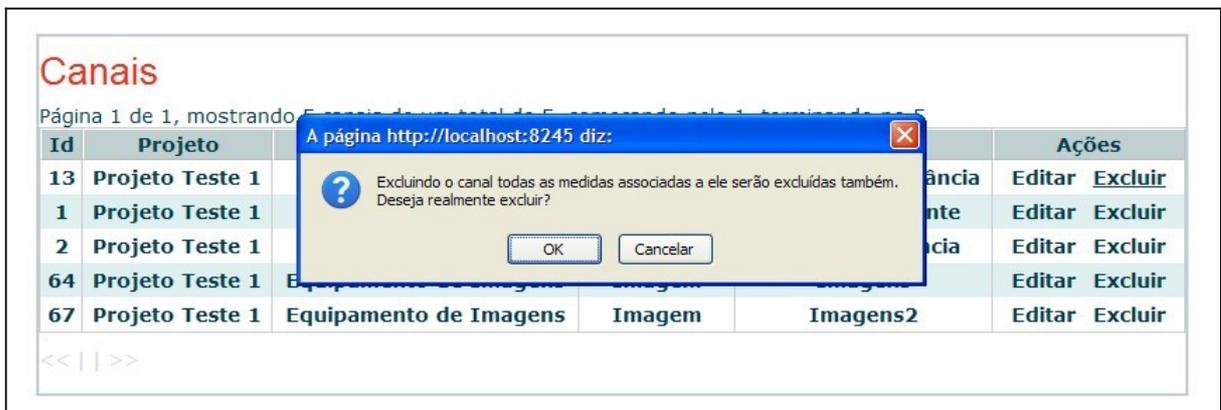


Figura 4-31: Confirmação de exclusão de um canal.

1.22.4.e - Baixar CSV

Na Figura 4 -32 é possível ver a janela de confirmação de *download* do arquivo que contém todas as medidas deste canal. O arquivo criado é do formato mostrado na Tabela 4 -8.

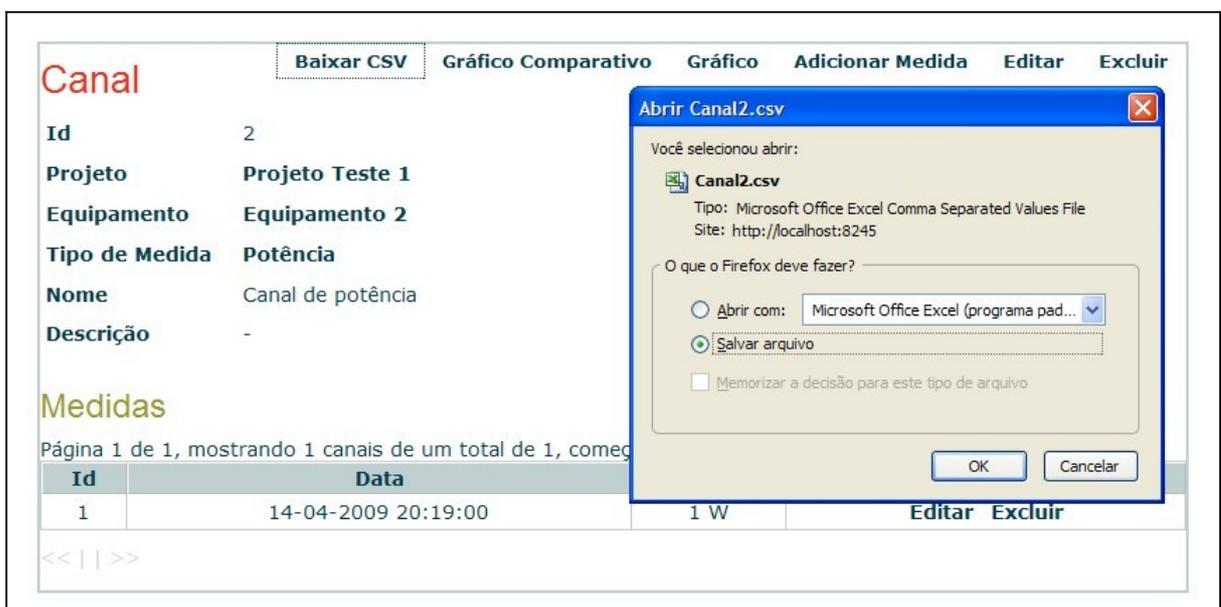


Figura 4-32: *Download* de arquivo, no formato CSV, de um canal.

Equipamento	Equipamento 2 (id 2)
Canal	Canal de potência (id 2)
Tipo	Potência (id 5, unidade W)
Data medida	Valor
14/4/2009 20:19:00	1

Tabela 4-8: Arquivo CSV criado para um canal.

1.22.4.f - Gerar Gráfico

Ao clicar nesta opção, um formulário é mostrado ao usuário, como ilustrado pela Figura 4 -33, para que possa ser feita a escolha dos parâmetros adicionais, que são: a data inicial; a data final; e o número de casas decimais das medidas. Após o envio deste formulário, o gráfico do canal é mostrado, conforme ilustrado na Figura 4 -34.

Escolha dos Parâmetros Adicionais

Data Inicial
 - - : :

Data Final
 - - : :

Casas Decimais

Figura 4-33: Formulário de escolha dos parâmetros adicionais.

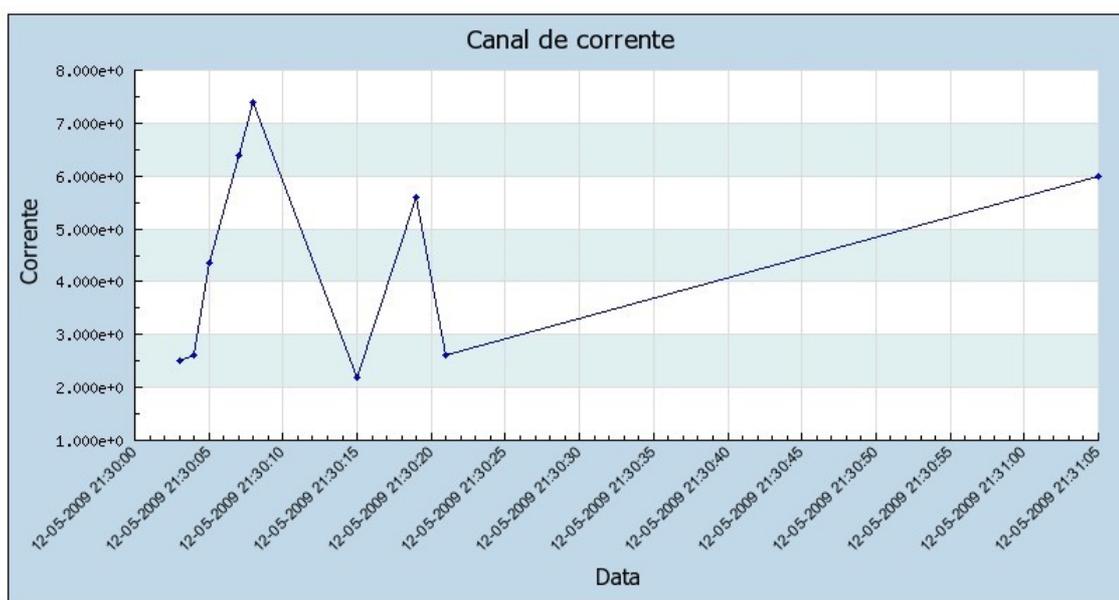


Figura 4-34: Gráfico de um canal.

1.22.4.g - Gerar Gráfico Comparativo

Ao clicar nesta opção, um formulário é mostrado ao usuário, como ilustrado pela Figura 4 -35, para que possa ser feita a escolha dos parâmetros adicionais, que são: o segundo canal utilizado na comparação das medidas; a data inicial; a data final; e o número de casas decimais das medidas. Após o envio deste formulário, o gráfico comparativo entre os dois canais escolhidos é mostrado, conforme ilustrado na Figura 4 -36.

Escolha dos Parâmetros Adicionais

Equipamento do Segundo Canal
Equipamento 2

Canal
Canal de potência

Data Inicial
12 - 05 - 2009 21 : 25 : 03

Data Final
12 - 05 - 2009 21 : 31 : 05

Casas Decimais
3

Figura 4-35: Formulário de escolha dos parâmetros adicionais.

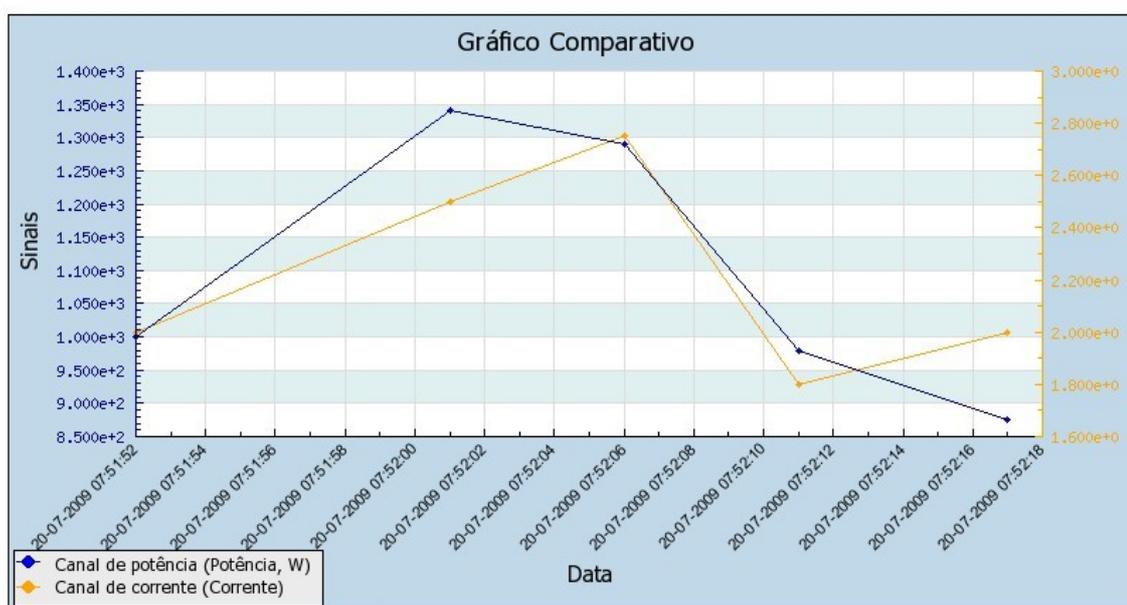


Figura 4-36: Gráfico comparativo entre dois canais.

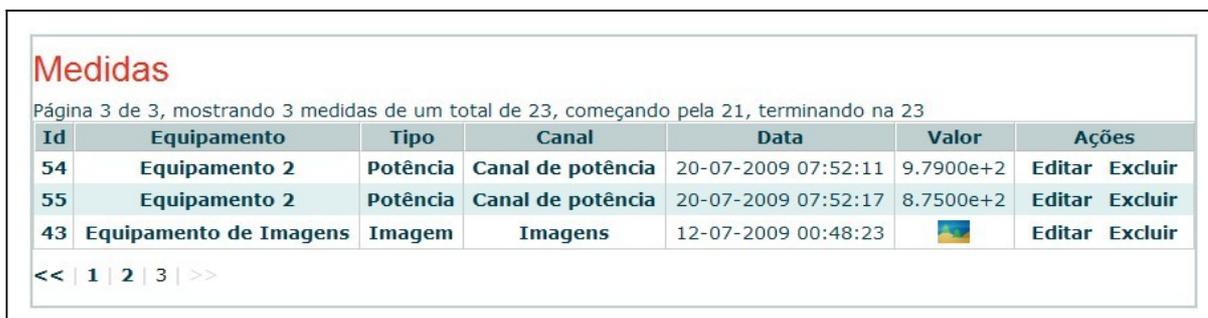
1.22.5 - Medidas

As funcionalidades do sistema relacionadas às medidas são listadas a seguir.

1.22.5.a - Listar Medidas

Pode-se observar pela Figura 4 -37, que o identificador, equipamento, tipo, canal, data e valor de cada medição são listados, além de serem dadas as opções de edição e exclusão de cada uma. Os nomes dos equipamentos, tipos de medida e canais possuem *links* para suas respectivas visualizações.

Para o caso do canal ser do tipo - Imagem -, ao invés de ser mostrado o valor da medida, é dado um *link* para a visualização da imagem Figura 4 -37.



Medidas						
Página 3 de 3, mostrando 3 medidas de um total de 23, começando pela 21, terminando na 23						
Id	Equipamento	Tipo	Canal	Data	Valor	Ações
54	Equipamento 2	Potência	Canal de potência	20-07-2009 07:52:11	9.7900e+2	Editar Excluir
55	Equipamento 2	Potência	Canal de potência	20-07-2009 07:52:17	8.7500e+2	Editar Excluir
43	Equipamento de Imagens	Imagem	Imagens	12-07-2009 00:48:23		Editar Excluir

<< | 1 | 2 | 3 | >>

Figura 4-37: Lista de medidas.

1.22.5.b - Adicionar/Editar Medida

As opções de inclusão e edição de medidas são idênticas, e permitem que cada campo de um registro da tabela de medidas seja alterado. A Figura 4 -38 mostra a edição de uma medida. Neste formulário de edição, o campo - Canal - varia de acordo com o - Equipamento - selecionado.

Editar Medida

Equipamento
 Equipamento 2

Canal
 Canal de potência

Data
 20 - 07 - 2009 07 : 52 : 11

Valor
 9.7900e+2

Enviar

Figura 4-38: Formulário de edição de uma medida.

1.22.5.c - Excluir Medida

Esta opção é dada na listagem de todas as medidas e na visualização do canal ao qual esta medida está associada. Ao clicar nesta opção, o usuário recebe uma mensagem de confirmação para a exclusão. Esta mensagem *pop-up* pode ser vista na Figura 4-39.

Canal

Baixar CSV Gráfico Comparativo Gráfico Adicionar Medida Editar Excluir

Id 2
Projeto Projeto Teste 1
Equipamento Equipamento 2
Tipo de Medida Potência
Nome Canal de potência
Descrição -

Medidas

Página 1 de 1, mostrando 5 canais de um total de 5, começando pelo 1, terminando no 5

Id	Data	Valor	Ações
51	20-07-2009 07:51:52	1.0000e+3 W	Editar Excluir

Figura 4-39: Confirmação de exclusão de uma medida.

Capítulo 5

1.23 - Resultados e Discussões

Os resultados do sistema estão relacionados aos testes realizados, levando-se em consideração todas as funcionalidades implementadas na interface gráfica. Ao fim do desenvolvimento individual de cada método das classes, foram feitos testes pontuais, passando parâmetros para os métodos, realizando troca de arquivos com um servidor FTP de teste, gravando no banco de dados, gerando gráficos etc. Todos estes testes obtiveram resultados satisfatórios no que diz respeito aos aspectos técnicos do sistema, uma vez que todas as funcionalidades foram executadas com sucesso.

A maior dificuldade encontrada no desenvolvimento do projeto foi a utilização do modelo de execução AJAX, que exigiu um grande esforço no aprendizado, para se obter resultados satisfatórios quanto à apresentação dos dados ao usuário.

Como teste final do sistema, foi realizada a configuração de um dos equipamentos do LIF, do projeto denominado - Vandalismo -, para que este atendesse aos padrões de transferência de dados do protocolo elaborado neste projeto. Este é um projeto de monitoramento por imagens de torres de transmissão em regiões remotas, contra vandalismos, com transmissão de dados via GSM (*Global System for Mobile communications*). Este teste seguiu os passos explicados a seguir.

Primeiramente foram adicionados ao banco de dados: o projeto - Vandalismo -; o protótipo - VandC -; e os canais - Imagens, Voltagens e Temperaturas -, responsáveis pelas medições feitas pelo equipamento instalado no laboratório. Estes dados podem ser vistos pela visualização do equipamento, na Figura 5 -40. Nesta figura são vistos também, os identificadores, automaticamente gerados pelo banco de dados, para o equipamento e para os canais de medida. Os campos - Domínio, DNS, Login, Senha e Diretório - do equipamento referem-se ao servidor FTP instalado no laboratório, que utiliza a porta padrão 21 para comunicações TCP/IP feitas através do protocolo FTP.

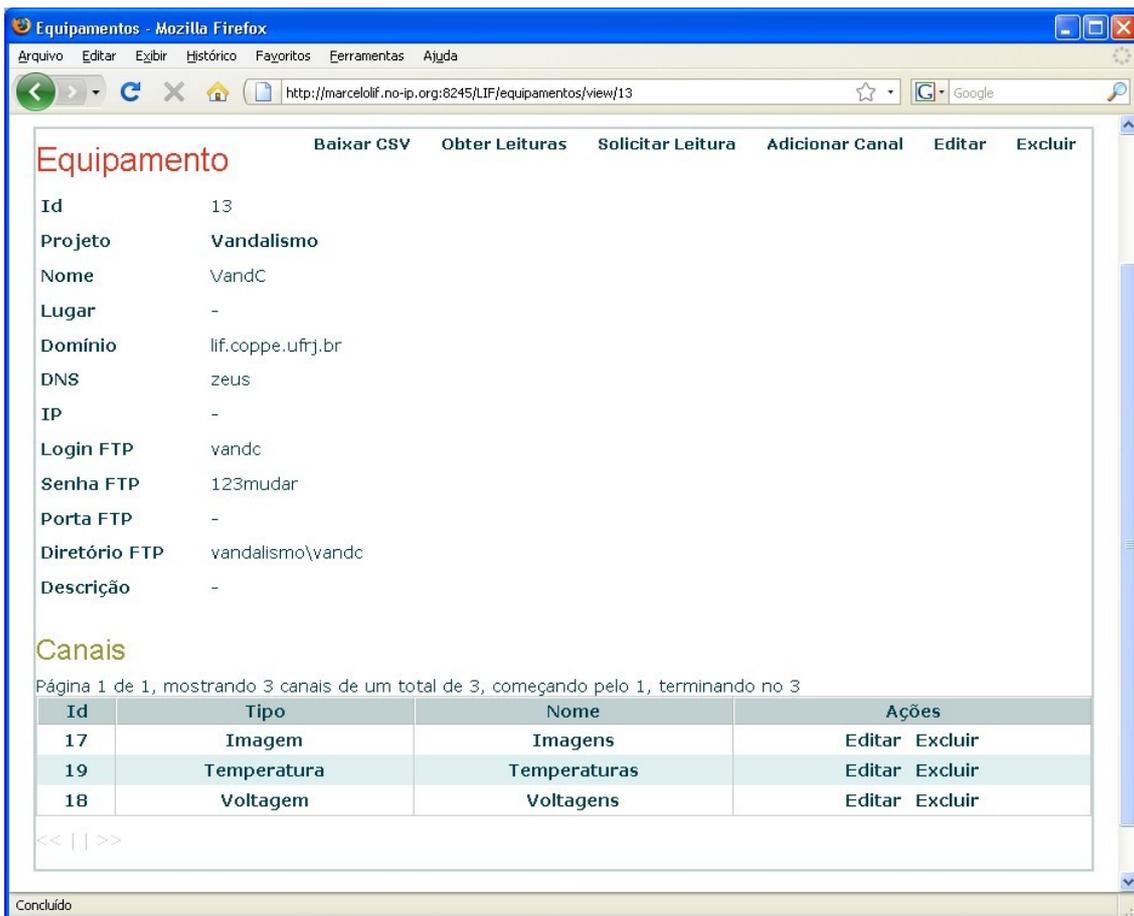


Figura 5-40: Visualização do equipamento - VandC - instalado no laboratório.

Após a inclusão destes dados, foram realizadas as medições, gerando arquivos, no formato CSV, com os dados obtidos. Um exemplo de arquivo gerado pode ser visto na Tabela 5 -9.

LIF-VANDC-20090721072400.DAT	21/7/2009 07:24	13	
2	17	21/7/2009 07:24	20090721072400_Temp_12_Bat_91.jpg
3	19	21/7/2009 07:24	22.8
4	18	21/7/2009 07:24	11.83

Tabela 5-9: Dados obtidos do equipamento - VandC -.

Esta tabela apresenta, em sua primeira linha, o cabeçalho do arquivo, com o nome do arquivo, a data de sua criação, e o identificador do equipamento. Nas demais linhas podem ser vistos os identificadores dos tipos de medições feitas, os identificadores dos canais, as datas das medidas, e os valores obtidos. No caso da imagem é mostrado o nome do arquivo criado.

Estes arquivos gerados pelo equipamento foram então obtidos, e a partir dos dados inseridos no banco de dados, obtiveram-se os gráficos de variação de temperatura, mostrado na Figura 5 -41, e de voltagem, mostrado na Figura 5 -42.

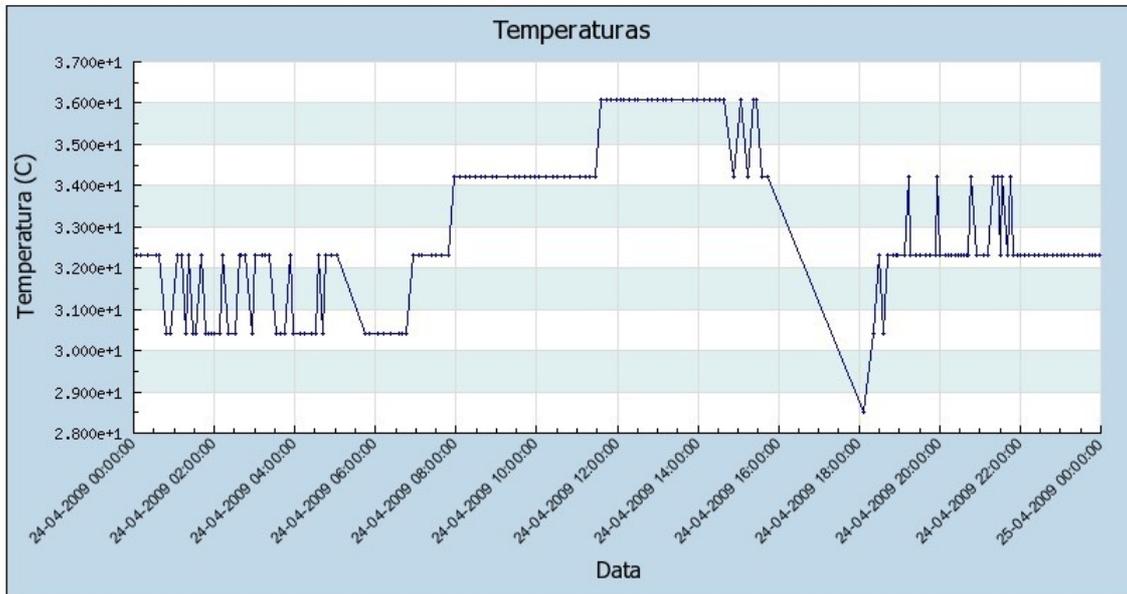


Figura 5-41: Gráfico de variação de temperatura para dados reais.

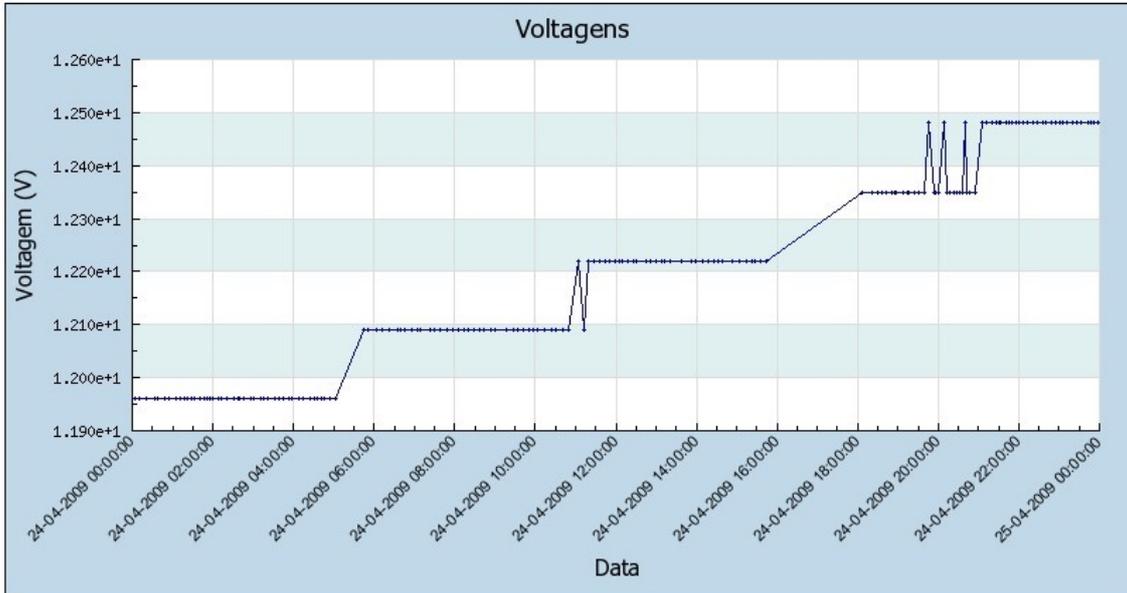


Figura 5-42: Gráfico de variação de voltagem para dados reais.

A visualização do canal de imagens do equipamento é mostrada na Figura 5-43.

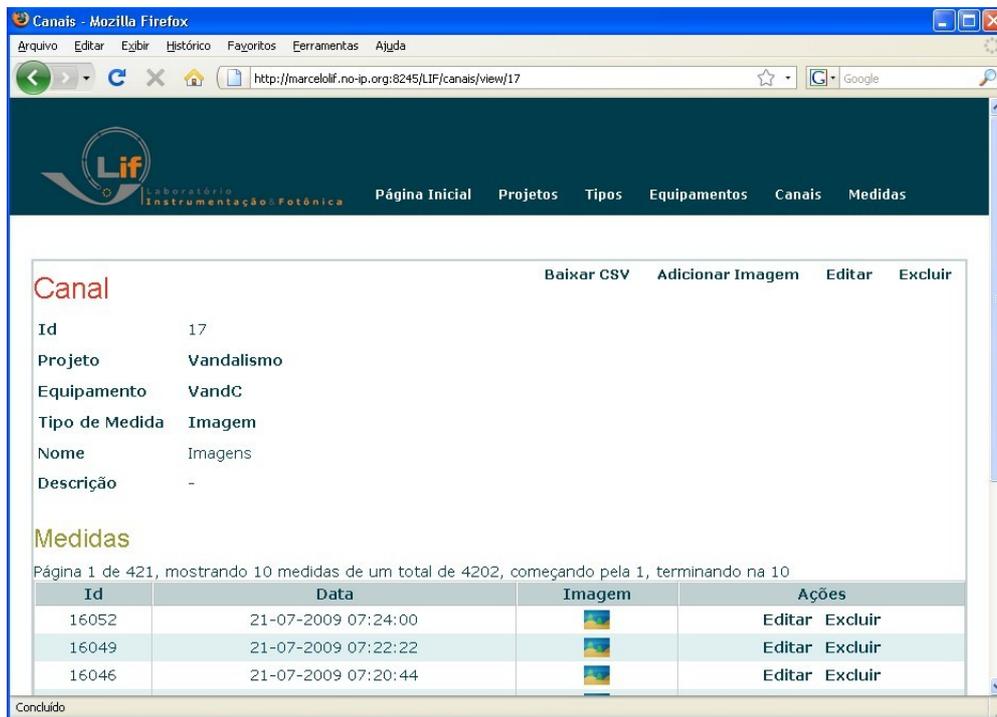


Figura 5-43: Visualização do canal - Imagens -.

Clicando-se em um dos *links* de imagem deste canal, obtém-se uma das imagens armazenada em um diretório específico para este canal no servidor, como ilustrado na .

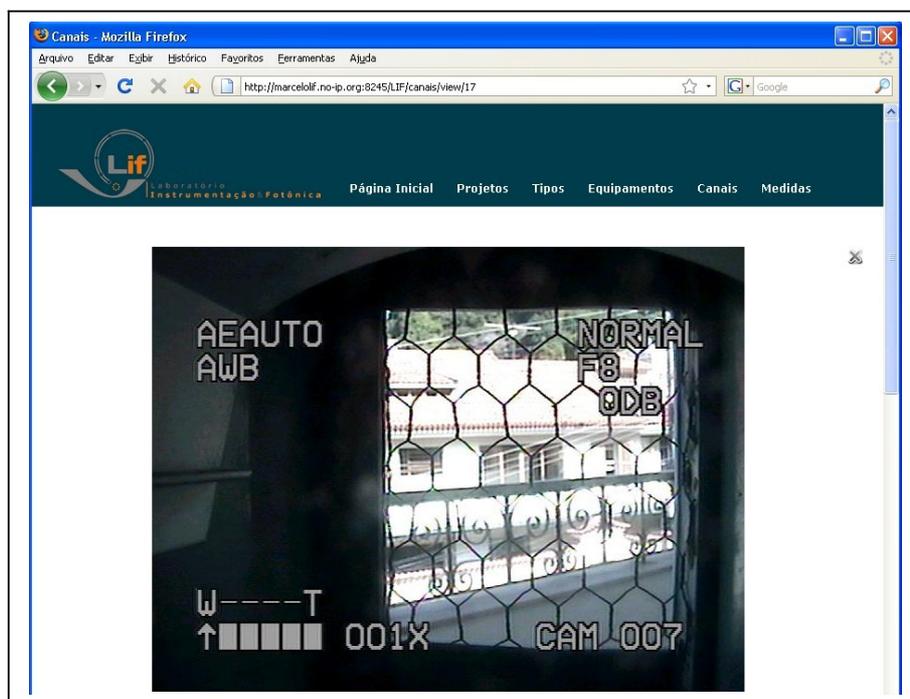


Figura 5-44: Imagem obtida pelo equipamento - VandC -.

Capítulo 6

1.24 - Conclusão

O LIF é um laboratório multidisciplinar que atua em projetos de pesquisa e desenvolvimento nas áreas de instrumentação e fotônica, e consultoria na área de instrumentação. Seus projetos, em parceria com empresas, possuem muitas medidas e informações que necessitam ser gerenciados. Este gerenciamento é indispensável, pois é através deste que toda análise e acompanhamento de projetos são realizados.

Neste projeto foi implementado um sistema *web* que estabeleceu um procedimento padrão de aquisição, gestão e análise dos dados recebidos pelo LIF, provenientes de seus protótipos de *hardware* de campo. Para tal, foram definidos um protocolo de transferência de dados e um modelo de banco de dados. Esta padronização possibilita uma redução no tempo e no custo do desenvolvimento de projetos que necessitam deste gerenciamento de dados.

O sistema obtém os dados através de conexões TCP/IP com os protótipos, que agem como servidores FTP, no modo passivo. Através dos dados adquiridos, é possível gerenciar a operação de um equipamento e de seus canais de leitura, além de monitorar as medidas adquiridas através de tabelas ou gráficos temporais. Estes dados podem ser exportados em arquivos, no formato CSV, para análises mais detalhadas. Este sistema também permite a inclusão, modificação e exclusão de projetos, protótipos, canais de medição e tipos de parâmetros armazenados no banco de dados.

Esse projeto contou com o apoio constante da equipe do LIF. O sistema encontra-se instalado no computador onde foi implementado. Durante o projeto, foram realizadas reuniões periódicas para compartilhar experiências e definir estratégias para seu desenvolvimento.

O sistema atendeu ao requisito de ser genérico o suficiente para suportar os mais variados tipos de projeto e de grandezas medidas pelos protótipos de *hardware*. De maneira geral, o sistema se comportou da maneira prevista, com sucesso, sem apresentar lentidões ou falhas em sua lógica programacional. A avaliação definitiva do

comportamento do sistema deverá ser realizada por seus futuros usuários, integrantes do LIF.

Como próximos passos do projeto, estão a migração para o servidor *web* do laboratório e a implementação da autenticação dos usuários, limitando o acesso e estabelecendo permissões à utilização das funcionalidades.

Os próximos equipamentos, desenvolvidos nos diferentes projetos do laboratório, devem obedecer ao protocolo estabelecido por este sistema, a fim de garantir a integração de todo o processo. Esta adoção do protocolo permitirá a independência do LIF em relação a terceiros no âmbito de telemetria.

Por fim, o sistema desenvolvido pode servir de base para outros, em diferentes laboratórios, pois em nada depende de características específicas do LIF, atendendo a equipamentos genéricos com atividades variadas. O desenvolvimento do projeto contribuiu de maneira significativa na agregação de conhecimento, o que será de grande valia para a vida profissional.

Bibliografia

- [1] RATZAN, L., “Understanding Information Systems: What They Do And Why We Need Them”. ALA Editions, 2004.
- [2] LABORATÓRIO DE INSTRUMENTAÇÃO E FOTÔNICA, <http://www.lif.coppe.ufrj.br/>, (Acesso em Abril 2009).
- [3] COPPE-UFRJ, “Instituto Alberto Luiz Coimbra de Pós-Graduação e Pesquisa em Engenharia”, <http://www.coppe.ufrj.br/>, (Acesso em Junho 2009).
- [4] CREATIVYST SOFTWARE, “The Comma Separated Value (CSV) File Format”, <http://www.creativyst.com/Doc/Articles/CSV/CSV01.htm>, (Acesso em Junho 2009).
- [5] PRESSMAN, R. S., “Software Engineering: A Practitioner’s Approach, McGraw-Hill Series in Computer Science”. 5 ed. McGraw-Hill, 2001.
- [6] MICROSOFT DEVELOPER NETWORK, “Model-View-Controller”, <http://msdn.microsoft.com/en-us/library/ms978748.aspx>, (Acesso em Abril 2009).
- [7] WORLD WIDE WEB CONSORTIUM, “RFC 959: File Transfer Protocol”, <http://www.w3.org/Protocols/rfc959/>, (Acesso em Julho 2009).
- [8] DATABASE JOURNAL, “Introduction to Relational Databases”, <http://www.databasejournal.com/sqlc/article.php/1469521/Introduction-to-Relational-Databases.htm>, (Acesso em Maio 2009).
- [9] OBJECT MANAGEMENT GROUP, “Unified Modeling Language: Superstructure”, <http://www.omg.org/docs/formal/07-02-03.pdf>, (Acesso em Julho 2009).

- [10] POSTGRESQL, <http://www.postgresql.org/>, (Acesso em Abril 2009).
- [11] ABOUT.COM, “The ACID Model”,
<http://databases.about.com/od/specificproducts/a/acid.htm>, (Acesso em Abril 2009).
- [12] INFO WESTER, “Conhecendo o Servidor Apache (HTTP Server Project)”,
<http://www.infowester.com/servapach.php>, (Acesso em Abril 2009).
- [13] NETCRAFT, “[May 2009 Web Server Survey](http://news.netcraft.com/archives/web_server_survey.html)”,
http://news.netcraft.com/archives/web_server_survey.html, (Acesso em Junho 2009).
- [14] CAKEPHP, <http://cakephp.org/>, (Acesso em Abril 2009).
- [15] PHP: Hypertext Preprocessor, <http://br.php.net/>, (Acesso em Julho 2009).
- [16] “Framework Design: A Role Modeling Approach”,
<http://dirkriehle.com/computer-science/research/dissertation/index.html>, (Acesso em Julho 2009).
- [17] ADAPTIVE PATH, “Ajax: A New Approach to Web Applications”,
<http://adaptivepath.com/ideas/essays/archives/000385.php> (Acesso em Junho 2009).
- [18] W3C, “Cascading Style Sheets”, www.w3.org/Style/CSS/, (Acesso em Maio 2009).
- [19] W3C, “Document Object Model”, <http://www.w3.org/DOM/>, (Acesso em Junho 2009).
- [20] W3C, “Hypertext Markup Language”, <http://www.w3.org/MarkUp/>, (Acesso em Julho 2009).
- [21] “JavaScript Source”, <http://javascript.internet.com/>, (Acesso em Junho 2009).