

UNIVERSIDADE DE TAUBATÉ

Ana Paula Müller Giancoli

**PROPOSTA DE SISTEMA PARA REGISTRO ELETRÔNICO
DE PONTO COM GERENCIAMENTO REMOTO**

Taubaté - SP

2011

Ana Paula Müller Giancoli

**PROPOSTA DE SISTEMA PARA REGISTRO ELETRÔNICO
DE PONTO COM GERENCIAMENTO REMOTO**

Dissertação apresentada para obtenção do Título de Mestre pelo Curso de Pós-Graduação do Departamento de Engenharia Mecânica da Universidade de Taubaté. Área de Concentração: Automação.

Orientador: Prof. Dr. Francisco Carlos Parquet Bizarria.

Taubaté - SP

2011

Giancoli, Ana Paula Müller

Proposta de sistema para registro eletrônico de ponto com gerenciamento remoto. / Ana Paula Müller Giancoli. Taubaté: Unitau – 2011.

85f.: il.

Dissertação (Mestrado). Curso de Engenharia Mecânica. Ênfase em Automação – Universidade de Taubaté.

Orientador: Francisco Carlos Parquet Bizarria

1. Portaria 1510. 2. Registrador Eletrônico. 3. Código Aberto.

ANA PAULA MÜLLER GIANCOLI

**PROPOSTA DE SISTEMA PARA REGISTRO ELETRÔNICO DE PONTO COM
GERENCIAMENTO REMOTO**

Dissertação apresentada para obtenção do
Título de Mestre pelo Curso de Pós-Graduação
do Departamento de Engenharia Mecânica da
Universidade de Taubaté.
Área de Concentração: Automação

Data: 31/janeiro/2011

Resultado: Aprovado

BANCA EXAMINADORA

Prof. Dr. Francisco Carlos Parquet Bizarria

UNITAU

Assinatura _____

Prof. Dr. José Carlos Lombardi

UNITAU

Assinatura _____

Prof. Dr. Carlos Henrique Netto Lahoz

DCTA/IAE

Assinatura _____

Dedico este trabalho à Deus, aos meus pais, filho e ao meu companheiro pelo apoio, força, incentivo, confiança e amizade. Sem eles nada disso seria possível.

AGRADECIMENTOS

Este trabalho estaria inacabado sem a compreensão de minha família e amigos, cujo apoio e incentivo influenciaram-me durante a sua confecção.

À Deus por me amparar nos momentos difíceis, me auxiliar a superar as dificuldades e mostrar o melhor caminho a trilhar.

Aos meus pais, Walter Giancoli Junior e Ediane Müller Giancoli, que sempre me mostraram o caminho da educação, sabedoria, amor e da verdade.

Ao meu filho, Gabriel, que com ternura e amor foi compreensivo e amigo durante dias e noites abstendo-se de brincadeiras e diversões em minha companhia.

Ao meu companheiro e amigo de todos os momentos de minha vida, Clayton Eduardo, que teve compreensão, carinho e amor, e me apoiou nos momentos difíceis tornando-os simples e fáceis.

À Helena pelos cafés e pelo auxílio nas questões administrativas e pessoais.

Ao professor Dr. João Bosco pelo apoio em momentos decisivos no início desta jornada.

Aos professores Dr. Carlos Lahoz e Dr. José Lombardi pelas sugestões e direcionamentos propostos durante o exame de qualificação.

Ao professor Dr. Francisco Carlos Parquet Bizarria pela orientação, paciência e apoio fornecido durante a confecção deste trabalho.

RESUMO

O presente trabalho propõe uma arquitetura de sistema para registro eletrônico de ponto baseado em *software* livre e capaz de atender os principais requisitos extraídos da portaria 1510 do Ministério do Trabalho. Essa arquitetura utiliza o sistema operacional Linux, a linguagem de programação Python, o framework Web Plone e o servidor de aplicações Zope, a fim de proporcionar, entre outros benefícios, a segurança, o acesso ao código fonte da aplicação e a independência de fornecedores. A validação é obtida por meio de testes práticos realizados em protótipo que adota os elementos do sistema. Os resultados satisfatórios obtidos nesses testes indicam que a aludida arquitetura é adequada para a aplicação em questão.

Palavras-Chave: Registrador Eletrônico de Ponto, Código Aberto, Plone, Portaria 1510.

ABSTRACT

This study aims at proposing a system architecture for electronic clocking in and out based on free software to fulfill the main requirements extracted from the 1510 regulation from the Ministry of Labor. This architecture uses the Linux operating system; the Python programming language; the Web Plone framework and the Zope application server to provide, among other benefits, the security, the access to application source code, and the independence from suppliers. The validation is obtained through practical tests on prototypes that adopt the elements of the system. The satisfactory results obtained in these tests indicate that the aforementioned architecture is suitable for the application in question.

Keywords: electronic clocking in and out, open code, Plone, 1510 regulation.

LISTA DE TABELAS

Tabela 1: Requisitos exigidos pela Portaria 1510 – primeira fase.....	33
Tabela 2: Requisitos exigidos pela Portaria 1510 – segunda fase	34

LISTA DE FIGURAS

Figura 1: Componentes da arquitetura proposta.....	30
Figura 2: Caso de Uso Gráfico para o acesso local	36
Figura 3: Caso de Uso Gráfico para o acesso remoto	37
Figura 4: Diagrama de Classes	46
Figura 5: Equipamentos utilizados nos testes práticos	47
Figura 6: Arquitetura do protótipo.....	48
Figura 7: Fluxograma de acesso local	51
Figura 8: Fluxograma de acesso remoto.....	57
Figura 9: Hierarquia de Interfaces disponíveis no sistema – módulo remoto	58
Figura 10: Hierarquia de Interfaces disponíveis no sistema – módulo local.....	59
Figura 11: Interface gráfica do sistema de registro eletrônico de ponto.....	60
Figura 12: Interface gráfica do sistema remoto – interface inicial	60
Figura 13: Interface gráfica do sistema remoto – menu principal.....	61
Figura 14: Interface gráfica do menu cadastros.....	62
Figura 15: Interface gráfica do cadastro de tipos de usuários	62
Figura 16: Interface gráfica de edição de tipos de usuários	63
Figura 17. Interface gráfica do menu de consultas.....	64
Figura 18. Interface gráfica utilizada para consultar os tipos de usuários.....	65
Figura 19. Interface gráfica do menu de relatórios.....	65
Figura 20: Informações de usuários registradas na memória de trabalho	66
Figura 21: Informações registradas na memória permanente.....	67
Figura 22: Interface gráfica de confirmação de registro eletrônico de ponto.....	67
Figura 23: Comprovante de registro eletrônico de ponto	68
Figura 24: Interface gráfica do sistema de registro - acesso remoto	68
Figura 25: Interface gráfica para geração dos arquivos da fiscalização	70

LISTA DE ABREVIATURAS E SIGLAS

AFD	Arquivo Fonte de Dados
BIO.	Leitor Biométrico de digital
CLT	Consolidação das Leis de Trabalho
COMP. LOCAL	Computador Local
COMP. REMOTO	Computador Remoto
CNPJ	Cadastro Nacional de Pessoa Jurídica
COD.	Leitor de Código de Barras
CPF	Cadastro de Pessoa Física
GPL	<i>General Public Lincese</i>
INT.	Internet
IP	<i>Internet Protocol</i>
IPH.	Telefone Celular Iphone
MRP	Memória de Registro de Ponto
MT	Memória de Trabalho
MTE	Ministério do Trabalho e Emprego
NOTE.	<i>Notebook</i>
PDF	<i>Printable Document Format</i>
PHP	<i>Hypertext Preprocessor</i>
PIS	Programa de Integração Social

RAD	<i>Rapid Application Development</i>
RDBMS	<i>Relational Database Management System</i>
REP	Registrador Eletrônico de Ponto
RLOC.	Rede Local
ROT.	Roteador
SERV.	Servidores
SGBD	Sistema Gerenciador de Banco de Dados
SQL	<i>Structured Query Language</i>
SREP	Sistema de Registro Eletrônico de Ponto
TCP	<i>Transmission Control Protocol</i>
UC	<i>Use Case</i>
UML	<i>Unified Modeling Language</i>
UPS.	<i>Uninterruptible Power Supply</i>
USB	<i>Universal Serial Bus</i>
WEB	<i>World Wide Web</i>

SUMÁRIO

CAPÍTULO 1. INTRODUÇÃO	15
1.1 Descrição do Problema	15
1.2 Solução Proposta	16
1.3 Objetivos.....	17
1.4 Revisão Bibliográfica	17
1.5 Estrutura do Trabalho	19
CAPÍTULO 2. CONCEITOS BÁSICOS.....	21
2.1 Portaria número 1510, de 21 de agosto de 2009	21
2.2 Modelagem de <i>software</i>	22
2.3 <i>Software</i> Livre	23
2.3.1 Licença GPL.....	24
2.4 Python, Zope e Plone.....	24
2.5 Redes	26
2.5.1 Conectividade	27
2.5.2 Protocolo de comunicação.....	28
CAPÍTULO 3. DESENVOLVIMENTO	30
3.1 Arquitetura física proposta	30
3.2 Requisitos de <i>hardware</i> e <i>software</i>	32
3.3 Arquitetura lógica	35
3.3.1 Requisitos exigidos pela portaria e implementados pelo trabalho	35
3.3.2 Casos de uso gráficos	36
3.3.3 Casos de uso textuais.....	37
3.3.3.1 Casos de uso do módulo local	37
3.3.3.1.1 UC Efetuar <i>login</i>	38
3.3.3.1.2 UC Validar acesso.....	38
3.3.3.1.3 UC Registrar entrada/saída.....	38
3.3.3.1.4 UC Gerar comprovante.....	39
3.3.3.1.5 UC Ajustar relógio.....	39
3.3.3.2 Casos de uso do módulo remoto.....	39
3.3.3.2.1 UC Efetuar <i>login</i>	40
3.3.3.2.2 UC Validar acesso.....	40
3.3.3.2.3 UC Manter tipos de usuários	40
3.3.3.2.4 UC Manter tipos de operações	41
3.3.3.2.5 UC Manter tipos de identificação.....	42
3.3.3.2.6 UC Manter tipos de identificação de registros	42
3.3.3.2.7 UC Manter grupos de usuários	43
3.3.3.2.8 UC Manter usuários.....	43
3.3.3.2.9 UC Enviar mensagem.....	44
3.3.3.2.10 UC Gerar arquivos da fiscalização	44
3.3.3.2.11 UC Gerar espelho de ponto.....	45
3.3.4 Diagrama de classe	46
3.4 Protótipo	47
3.4.1 Arquitetura do protótipo	47
3.4.2 Programa de gerenciamento	49

3.4.3	Interface gráfica.....	58
3.4.4	Testes funcionais.....	66
3.4.4.1	Acesso local e operação dedicada	66
3.4.4.2	Acesso remoto e módulo de gerenciamento	68
3.4.5	Resultados.....	70
CAPÍTULO 4. CONCLUSÕES E TRABALHOS FUTUROS.....		71
REFERÊNCIAS		73
APÊNDICE A – Codificação.....		75

CAPÍTULO 1. INTRODUÇÃO

Este capítulo é constituído pela descrição do problema, pela solução proposta, pelos objetivos do trabalho, pela revisão bibliográfica e pela estrutura do trabalho. A descrição do problema destaca os fatores que motivaram a elaboração do trabalho proposto. Na solução proposta é apresentada uma arquitetura de sistema para efetivar o registro eletrônico de ponto conforme previsto na portaria do Ministério do Trabalho e Emprego (MTE). Nos objetivos do trabalho é estabelecido que a apresentação de um sistema para registro eletrônico de ponto e sua avaliação por meio de testes práticos são as metas principais deste trabalho. A revisão bibliográfica apresenta os trabalhos pesquisados e que guardam relação com a presente dissertação. Na estrutura do trabalho é apresentada a estrutura organizacional utilizada na confecção dessa dissertação com os respectivos resumos.

1.1 Descrição do Problema

Segundo BRASIL (2010), o procedimento de controle de frequência é amplamente utilizado por empresas de todo o mundo para registro dos horários de entrada e saída de seus funcionários. Esse tipo de registro é importante para o empregador e para seus colaboradores, uma vez que garante para ambos o cumprimento dos direitos e deveres das partes envolvidas. Existem basicamente três tipos principais de registradores de ponto: o manuscrito, os baseados em registradores mecânicos e os eletrônicos. Os dois primeiros, apesar de estarem tecnologicamente defasados, ainda são muito utilizados, em especial por empresas de pequeno porte. Os principais limites da adoção do tipo manuscrito e do tipo mecânico nos dias atuais são: a precariedade com que as informações são armazenadas e a facilidade com que esses podem ser burlados, em função da fragilidade dos métodos e mecanismos empregados. Além disso, essa metodologia se resume a somente registrar o horário de entrada e saída dos funcionários, sem nenhuma possibilidade de expansão de funcionalidades, tratamento de dados ou mesmo cópia de segurança. O registrador eletrônico de ponto por sua vez, é muito mais moderno e possui mecanismos de segurança que dificultam a alteração de informações e registros, tornando a informação armazenada muito mais confiável do que as obtidas nos

outros dois métodos. Até pouco tempo atrás, os fornecedores de registrador eletrônico de ponto, que neste trabalho será referenciado por registrador eletrônico de ponto (REP), tinham total autonomia para desenvolver seus sistemas de *hardware*¹ e *software*², agregando funcionalidades de maneira indiscriminada. No entanto, no dia 21 de agosto de 2009, o Ministério do Trabalho e Emprego (MTE), emitiu a portaria 1510, que regulamenta a utilização e a produção de registradores eletrônicos de ponto. De acordo com a portaria, registradores eletrônicos de ponto que estejam em funcionamento, deverão obrigatoriamente possuir especificações de *software* e *hardware* particulares, compatíveis com as especificadas em sua descrição. Registradores eletrônicos de ponto que porventura estiverem em uso sem a devida adaptação serão considerados ilegais, podendo a empresa que os utiliza ser penalizada por meio da aplicação de multas e processos judiciais. Cabe mencionar que os registradores mecânicos e sistemas de ponto manuscritos podem continuar sendo utilizados sem nenhum tipo de adaptação.

Com a publicação dessa portaria, o Ministério do Trabalho e Emprego (MTE) passou a regulamentar as especificações dos Registradores Eletrônicos de Ponto (REP) e dar ênfase ao registro fiel das marcações efetuadas, a existência de uma mídia local de armazenamento independente de conexão remota, a emissão de comprovante para cada registro de ponto e a existência de um ponto de comunicação padrão *Universal Serial Bus* (USB), utilizada para a transferência dos dados armazenados no REP em eventuais fiscalizações realizadas por auditores fiscais do trabalho. Desse modo, funções que outrora estariam incorporadas aos REP's, passarão a ser implementadas nos Sistemas de Registro Eletrônico de Ponto (SREP), que, quando utilizados, deverão ser instalados em servidores específicos independentes dos REP's.

1.2 Solução Proposta

Adotar um sistema baseado em *software* livre com estrutura capaz de atender as principais prescrições da portaria de número 1510 do Ministério do Trabalho e Emprego (MTE) e que permita a expansão e independência de fornecedores em adequações futuras.

¹ *Hardware* - conjunto de elementos físicos de um computador e os dispositivos a ele relacionados, como parte mecânica, magnética, elétrica e eletrônica. Refere-se à capacidade física da máquina.

² *Software* - conjunto de todos os recursos humanos, lógicos e mesmo de instalação e de organização, com os quais se explora uma máquina, equipamento ou sistema.

1.3 Objetivos

Este trabalho tem como objetivos: i) propor uma arquitetura de sistema, baseada em *software* livre, para efetuar o registro eletrônico de ponto em conformidade com o previsto na portaria do Ministério do Trabalho e Emprego (MTE) de número 1510, datada de 21 de agosto de 2009 e, ii) validar as funcionalidades dessa arquitetura por meio de testes práticos em protótipo que adote os principais blocos desse sistema.

1.4 Revisão Bibliográfica

Em pesquisas realizadas no ano de 2010, foram observados vários artigos técnicos que guardam relação com o presente trabalho, sendo os mais relevantes apresentados nesta subseção, porém, não foram encontrados registros de trabalhos acadêmicos focados no desenvolvimento do tema em questão de forma direta.

Atualmente, existem 81 registradores eletrônico de ponto (REP) distintos homologados pelo Ministério do Trabalho e Emprego (MTE, 2010). Segundo G1 (2010), as soluções apresentadas custam de R\$ 2 mil a R\$ 7 mil em média. Todas as soluções homologadas são baseadas em *software* e *hardware* proprietários e substituem soluções disponíveis até o momento. Após a publicação desta portaria de número 1510, essas soluções, agora descontinuadas, exigem dessa forma, um novo investimento, de valor considerável, por parte dos utilizadores desse tipo de mecanismo de registro de ponto.

Até o presente momento, por meio das pesquisas bibliográficas não foi identificada nenhuma aplicação livre de *hardware* ou *software* homologada, mesmo porque, para que o procedimento seja realizado, obrigatoriamente deve existir uma solução completa disponível (*hardware* + *software*).

Durante o processo de revisão bibliográfica, foi possível encontrar diversos sistemas livres de registro eletrônico de ponto. Nenhuma entretanto, compatível com a portaria de número 1510. As principais são listadas à seguir:

Segundo PAPIZAN (2006), o PHP Timeclock, como o próprio nome sugere, é desenvolvido em linguagem PHP³ e voltado para *web*⁴. O projeto teve início em 2006 e não possui versões recentes, pois é um sistema norte-americano/EUA e, por esse motivo, não considera a normatização brasileira de leis trabalhistas. Como é um projeto antigo, é citado como referência por outros projetos, como por exemplo o Sispono.

De acordo com MONTEIRO (2008), a aplicação Conep é completa, pois integra uma solução de *software* a uma implementação de *hardware* de custo baixo e compatível com padrões livres. O Conep é desenvolvido utilizando o GAMBAS-2⁵ de uso semelhante a uma alternativa utilizada comercialmente. Infelizmente a aplicação proposta pelo autor é datada de meados de 2008 e, por esse motivo, não atende aos requisitos exigidos pela portaria número 1510. Nesse sentido, não foi observado atualização do projeto.

Conforme AGUIAR (2008), o S1 Ponto é desenvolvido em linguagem Python e funciona em uma interface baseada em *desktop*. É um projeto validado, mas não foi encontrada nenhuma menção ao *software* e conseqüentemente versões atualizadas que poderiam atender à portaria 1510, no site de seu desenvolvedor.

Segundo LEAL (2008), o Sispono é mais um sistema web desenvolvido em linguagem PHP, que segundo seu autor, teve na simplicidade do PHP Timeclock, o maior incentivo para ter sido desenvolvido. Cabe mencionar que além das informações apresentadas, outras poderão ser encontradas no arquivo LEIAME.txt, presente na versão 0.1 do Sispono. A última, e única versão disponível, é datada de meados de 2008 e não há menção a nenhuma atualização no *site* do projeto.

De acordo com FUZZER (2009a), o wPonto é um *fork*⁶ do projeto Sispono e seu desenvolvimento foi motivado, segundo seu autor, pela inexistência de um método de controle de versão no projeto original, dificultando dessa forma, o gerenciamento de contribuições e correção de *bugs*. A versão atual é a 1.1.3, lançada em meados de 2009, no entanto, existe no *blog* do autor, FUZZER (2009b), uma menção à versão 2.0, ainda em desenvolvimento, que a princípio possui somente correção de *bugs* encontrados nas versões

³ PHP: *Hypertext Preprocessor*, é uma linguagem de programação interpretada utilizada para desenvolvimento para *web*.

⁴ *Web*: uma página *web*, também é conhecida pelo equivalente em inglês *webpage*, e significa uma página na rede de alcance mundial.

⁵ GAMBAS-2: é um ambiente de desenvolvimento visual baseado em linguagem *Basic*.

⁶ *Fork*: em Engenharia de *software*, o *fork* de um projeto ocorre quando uma cópia legalizada do código fonte de um conjunto de *softwares* é realizada e um processo de desenvolvimento independente dessa é iniciado caracterizando assim, uma versão alternativa à original.

anteriores, sem adição de nenhuma funcionalidade relevante. Essa observação entretanto, leva a crer que o projeto continua em andamento, ainda que com atualizações demoradas.

Conforme WAGNER (2010), o sistema FingerFX2 é desenvolvido parcialmente em GAMBAS-2, interface gráfica, e em linguagem C++, em especial as rotinas de comunicação com o leitor biométrico e demais *drivers*⁷ externos. Possui suporte a cadastro com foto capturada diretamente da *webcam*⁸ e utiliza o banco de dados *Sqlite*⁹. Suporta também dispositivos *touchscreen*¹⁰. Nesse sentido, esse *software* suporta o maior número de dispositivos de entrada e saída, comparando-se em funcionalidade inclusive aos concorrentes de maior custo baseados no modelo de desenvolvimento proprietário, no entanto, não atende aos requisitos exigidos pela portaria número 1510. Observou-se que no *site* do desenvolvedor não há nenhuma menção a qualquer tipo de atualização que possibilite essa adequação.

1.5 Estrutura do Trabalho

O Capítulo 1: Introdução é composto pela descrição do problema, pela solução proposta, pelos objetivos do trabalho e por uma revisão bibliográfica. Na descrição do problema são apresentados os principais itens citados na portaria de número 1510 do Ministério do Trabalho e Emprego e os diferentes mecanismos de registro de ponto, dentre eles, o eletrônico. Na solução proposta é apresentada uma arquitetura de sistema capaz de atender os principais itens dessa portaria. Nos objetivos do trabalho são determinadas as principais metas que deverão ser alcançadas com o desenvolvimento deste trabalho, ou seja, a elaboração de proposta de arquitetura de sistema e sua validação por meio de testes em protótipo. Na revisão bibliográfica, são relacionados os sistemas mais relevantes de registros eletrônicos de ponto existentes que guardam relação com o tema desta dissertação.

No Capítulo 2: Conceitos Básicos, são descritos os principais requisitos da portaria 1510, os conceitos básicos da infra-estrutura de *hardware* proposta para o desenvolvimento

⁷ *Drivers*: são pequenos programas ou rotinas usados para interfacear e gerenciar um dispositivo de entrada e saída.

⁸ *Webcam*: é um dispositivo para captura de imagens utilizado principalmente na internet.

⁹ *Sqlite*: é uma biblioteca em linguagem C que implementa um banco de dados SQL embutido. Programas que usam a biblioteca SQLite podem ter acesso a banco de dados SQL sem executar um processo *Relational Database Management System* – sistema de gerenciamento de banco de dados relacional (RDBMS) separado.

¹⁰ *Touchscreen*: é um tipo de tela presente em diferentes equipamentos, sensível a toque e que por isso dispensa o uso de equipamentos como teclados e mouses.

do trabalho, as linguagens de programação utilizadas para a elaboração do programa de gerenciamento e registro eletrônico de ponto, a plataforma e *framework*¹¹ utilizados para desenvolvimento do protótipo de *software*. O entendimento desses conceitos substanciará a base para o leitor deste trabalho compreender as definições, os termos, os conceitos e os componentes utilizados nesta dissertação.

No Capítulo 3: Desenvolvimento, é apresentado o diagrama representativo da arquitetura proposta e a descrição de seus componentes. Entretanto, apresenta a modelagem do *software*, o protótipo desenvolvido e as interfaces gráficas, o fluxograma analítico de funcionamento, o detalhamento dos testes realizados e os seus resultados obtidos.

No Capítulo 4: Conclusões e Trabalhos Futuros, são apresentados as conclusões obtidas no desenvolvimento do trabalho, e as propostas para trabalhos futuros a partir dos elementos envolvidos nesta dissertação.

No Capítulo 5: Referências, são apresentados pesquisas bibliográficas que guardam relação com o desenvolvimento do presente trabalho.

No Apêndice A – Codificação: estão incluídos seções do código fonte da aplicação desenvolvida. Sendo que as demais codificações são inerentes ao gerenciador de conteúdo.

¹¹ *Framework*: um *framework*, em desenvolvimento de software, é uma abstração que une códigos comuns entre vários projetos de software provendo uma funcionalidade genérica. Um *framework* pode atingir uma funcionalidade específica, por configuração, durante a programação de uma aplicação. Ao contrário das bibliotecas, é o *framework* quem dita o fluxo de controle da aplicação.

CAPÍTULO 2. CONCEITOS BÁSICOS

Este capítulo tem como objetivo principal apresentar de forma resumida, visando à aplicação executada neste trabalho, os conceitos sobre a portaria de número 1510, a modelagem de *software*, o *Software Livre*, a Licença GPL, a Linguagem de Programação Python, o servidor de aplicação Zope e gerenciador de conteúdo Plone. Esses conceitos formarão a base mínima necessária para compreensão das terminologias e sistemas utilizados nesta dissertação.

2.1 Portaria número 1510, de 21 de agosto de 2009

No dia 21 de agosto de 2009, o Ministério do Trabalho e Emprego (MTE) publicou a portaria número 1510 que tem como objetivo disciplinar o registro eletrônico de ponto e a utilização do Sistema de Registro Eletrônico de Ponto (MTE, 2009). Nessa portaria, composta por 31 artigos, as características necessárias para a produção do *hardware* dos registradores eletrônicos de ponto e implementação dos *softwares* associados ao mesmo são detalhadas. De maneira geral, a portaria estabelece padrões e normativas que garantem o cumprimento dos direitos e deveres do empregador e de seus colaboradores, bem como a confidencialidade, integridade e disponibilidade das informações armazenadas nesse tipo de dispositivo.

O trabalhador não pode por exemplo, ser impedido de registrar o ponto. O equipamento deve, obrigatoriamente, fornecer a seu utilizador comprovantes individuais de marcação de ponto logo após o registro, tanto no procedimento de entrada como no de saída. Arquivos pré-formatados específicos para fiscalização também devem estar disponíveis em ocasiões específicas, utilizando-se para tanto, uma porta padrão USB externa.

De maneira geral, a portaria descreve com detalhes os pontos anteriormente mencionados, entre outros, bem como as particularidades de cada um. Desse modo, a leitura na íntegra da portaria, disponível gratuitamente na internet, é recomendada.

É importante salientar que a utilização de dispositivos eletrônicos de registro de ponto não é obrigatória e nem é imposta pela portaria. No entanto, todas as empresas que atualmente utilizam esse tipo de sistema eletrônico e tem contratos de trabalho e registros normatizados

pela Consolidação das Leis de Trabalho (CLT) possuem o prazo de um ano, a partir da data de publicação da portaria, para se adequar e continuar a fazer uso desse tipo de tecnologia. Na hipótese de não atenderem esses requisitos estarão ilegais e poderão ser penalizadas de acordo com a legislação vigente.

A data final para adequação dos registradores e sistemas de registro eletrônico aos requisitos da portaria, em princípio, ocorreria em 26 de agosto de 2010, no entanto, em função de dificuldades técnicas de implementação e conseqüente impossibilidade de produção de registradores eletrônicos de ponto (REP) em tempo hábil, para toda a comunidade de usuários, estimada em 700 mil empresas, o Ministério do Trabalho e Emprego prorrogou o prazo para 01 de março de 2011.

2.2 Modelagem de *software*

Segundo BOOCH (2005), a construção de modelos para compreender melhor o sistema que está sendo desenvolvido é uma prática que deve ser aplicada em diversas áreas, pois a modelagem de um *software* é a simplificação da realidade, de forma que os modelos fornecem uma cópia do projeto de um sistema. Com a modelagem, é possível visualizar o sistema como ele é, permitir especificar a estrutura ou o comportamento do mesmo, proporcionar um guia de construção, e documentar decisões tomadas. A visão tradicional no desenvolvimento de *software* adota a perspectiva de um algoritmo. Porém, a visão contemporânea no desenvolvimento de *software* adota uma perspectiva orientada a objetos, de forma que o principal bloco de construção de todos os sistemas é o objeto ou a classe. Todos os objetos têm uma identidade, um estado e um comportamento. Sendo assim, o método orientado a objetos para desenvolvimento de *software* é uma parte do fluxo principal, simplesmente porque tem sido provado seu valor para a construção de sistemas em todos os tipos de domínios de problemas, abrangendo todos os graus de tamanho e de complexidade.

Para a elaboração de estruturas de projetos de *software*, BOOCH (2005) propôs a utilização de uma linguagem-padrão, conhecida como *Unified Modeling Language* (UML). A UML poderá ser utilizada para visualizar, especificar, construir e documentar desde sistemas simples até os mais complexos. É parte de um método para desenvolvimento de *software*. E

por meio de diagramas, representações gráficas, permite-se a visualização do sistema como todo sob diversas perspectivas.

Nesse trabalho, utilizou-se das técnicas de modelagem com UML para levantamento de requisitos ou necessidades de software, para a identificação das funcionalidades, sendo que essas são representadas pelos diagramas de caso de uso e posteriormente, o diagrama de classes.

2.3 *Software* Livre

A utilização de computadores, dispositivos eletrônicos e processos de automação industrial, em geral requerem a utilização de plataforma baseada em sistema operacional e *softwares* de gerenciamento, desenvolvidos com o uso de uma ou mais linguagens de programação.

A escolha da plataforma adequada, bem como de seus componentes, é um fator decisivo para o sucesso e continuidade de um projeto e deve ir além do imediatismo necessário para o desenvolvimento de uma aplicação, considerando também aspectos de segurança, documentação disponível, adequação com o objetivo proposto, preservação do investimento e sobretudo, o quesito de independência tecnológica. Em muitos casos, as empresas que adotam esse tipo de tecnologia não possuem pessoal técnico qualificado para o desenvolvimento de *software* e, por esse motivo, a terceirização das soluções adotadas torna-se necessária. Grande parte das empresas que desenvolvem *software*, individualmente ou associados a algum tipo de *hardware*, utilizam o modelo de desenvolvimento de *software* com código fechado, onde o desenvolvedor não fornece o código-fonte da aplicação ao cliente, impedindo que esse realize adequações no *software* e garantindo assim, contratos de manutenção do sistema, situação que torna o cliente dependente do fornecedor e de imprevistos e ocorrências que essa relação pode causar. Nos últimos anos entretanto, um novo modelo de desenvolvimento tem se tornado popular, o modelo de desenvolvimento baseado em *software* livre, conforme é abordado nas subseções seguintes.

2.3.1 Licença GPL

Um *software* é considerado livre quando possibilita ao seu usuário usufruir dos quatro tipos de liberdade de *software* definidas pela *Free Software Foundation*, (STALLMAN, 2010):

- A liberdade para executar o programa, para qualquer propósito.
- A liberdade de estudar como o programa funciona, e adaptá-lo para as suas necessidades.
- A liberdade de redistribuir, inclusive vender, cópias de modo a ajudar ao próximo.
- A liberdade de modificar o programa, e liberar essas modificações, de modo que toda a comunidade se beneficie.

Em muitos casos, existem projetos livres com código-fonte e documentação disponíveis na internet que atendem a um grande número de pessoas e necessidades. Nesse caso, apesar de se tratar de uma aplicação inicialmente sem custo, o usuário obrigatoriamente, precisa se adequar ao *software* e não o contrário. No entanto, a adoção de soluções baseadas em *software* livre permite ao usuário personalizar seu sistema por meio da implementação de suas próprias contribuições, atendendo dessa forma, às suas necessidades específicas. Essa contribuição pode ser implementada pelo próprio usuário, ou por sua equipe de desenvolvimento, caso exista essa habilidade/possibilidade, ou ainda, pode ser repassada a um desenvolvedor terceirizado da preferência do usuário, caso o desenvolvedor original, por algum motivo, deixar de ser a melhor opção. Nesse caso específico, o desenvolvedor irá receber somente pelo trabalho realizado, sendo enquadrado assim, no modelo tradicional de prestador de serviços, garantindo ao usuário todas as liberdades citadas anteriormente.

2.4 Python, Zope e Plone

Atualmente, existem diversas linguagens de programação disponíveis, bem como, vários paradigmas de desenvolvimento, plataformas de sistemas operacionais e arquiteturas de

hardware. No entendimento do autor, tecnologias atuais, tipicamente utilizadas e com vasta documentação têm maior chance de sucesso, pois essas características tendem a perpetuar sua aceitação e conseqüente continuidade, bem como, atrair novos usuários.

Aplicativos e soluções de *software* desenvolvidos para ambiente *web* ou para *intranets* são comuns. As razões para tanto, são várias: independência de clientes, tanto em relação ao navegador utilizado como ao sistema operacional utilizado, a variedade de dispositivos que possuem esse tipo de cliente disponível, a abrangência que esse tipo de tecnologia atinge, não restringindo a solução a somente um âmbito local entre diversos outros. As tecnologias utilizadas em ambiente de desenvolvimento *web* em geral, são multiplataforma, sendo compatíveis com diferentes arquiteturas de *hardware* e sistemas operacionais. No entanto, em grande parte dos casos, o desenvolvimento de uma aplicação voltada à internet ou intranet exige cuidados minuciosos relacionados à segurança. Para tanto, o programador precisa ter um cuidado especial com os elementos de segurança da aplicação, além de submetê-la a um conjunto considerável de testes antes de tornar disponível para uso.

Existem *frameworks* que oferecem facilidades nesse sentido, no entanto, em geral, exigem equipamentos robustos para funcionar adequadamente e um alto grau de especialização do desenvolvedor. Outras linguagens, ainda que bastante populares, oferecem riscos de segurança em função das vulnerabilidades existentes em sua implementação e/ou arquivos de iniciação. Por esses motivos, teve-se o cuidado de escolher uma linguagem de programação estável, segura, com vasta documentação, amplamente utilizada e sobretudo atual para o desenvolvimento deste trabalho.

Para atender as características anteriormente mencionadas, optou-se pela utilização da linguagem python. Essa foi desenvolvida por Guido Van Rossum no início da década de 90. A linguagem é de fácil entendimento, pode ser utilizada em diferentes paradigmas de desenvolvimento (programação funcional, procedural, orientada a objetos, orientada a aspectos), diferentes arquiteturas de *hardware*, diferentes sistemas operacionais e é compatível com diversos tipos de ambientes distintos (interfaces modo texto, modo gráfico, web, celulares e dispositivos móveis, entre outros). Uma outra vantagem é que python é uma linguagem que não exige *softwares* de desenvolvimento *Rapid Application Development* (RAD) para que possa ser utilizada, além de se fundamentar em boas práticas de programação, como por exemplo, a indentação. Uma outra vantagem que levou à escolha do python foi a disponibilidade de aplicações adicionais que agregam funcionalidades excelentes a esse tipo de ambiente. As mais utilizadas nesse trabalho para o gerenciamento de conteúdo e

armazenamento de aplicações web, são as aplicações Plone e Zope, ambas desenvolvidas respectivamente em python.

Aplicativos de gerenciamento de conteúdo como o Plone, são utilizados para que vários usuários possam interagir com um *site* ou aplicação *web* sem conhecimento aprofundado de programação. O desenvolvimento da aplicação, conforme necessidade dos usuários, requer programação por parte do desenvolvedor. Outro fator importante agregado por esse tipo de aplicativo é o suporte a métodos e procedimentos de segurança, por meio da utilização de sessões e métodos de criptografia. Desse modo, há um ganho de tempo, pois o programador não necessita desenvolver essa implementação, minimizando as dificuldades de elaboração das codificações exigidas. Já o servidor de aplicação, neste caso denominado Zope, armazena o gerenciador de conteúdo e as informações da aplicação nele contida. Além disso, possibilita a utilização de sistemas gerenciadores de banco de dados, que permitem que grandes volumes de dados sejam armazenados, consultados, alterados e removidos com eficiência e agilidade, complementando ainda mais a funcionalidade das aplicações. Para tanto, utiliza a linguagem padrão para esse tipo de finalidade, o SQL. O sistema gerenciador de banco de dados utilizado nesse trabalho é o MySQL (MYSQL, 2010), em função de sua diversidade de recursos, reconhecimento no mercado, tipo de licença de uso e tradição.

2.5 Redes

A utilização de computadores para auxiliar na resolução de problemas é uma tendência mundial que já vem sendo utilizada desde a década de 80. Em geral, a ajuda provida por esse tipo de equipamento é fornecida por meio do desenvolvimento de *softwares*, ou de programas, denominados aplicativos. Esses aplicativos por sua vez, têm funcionalidades específicas e são comercializáveis, ou ainda podem ser desenvolvidos por empresas especializadas, adequados às necessidades do cliente. O fato é que, de maneira geral, as empresas possuem um número reduzido de computadores.

Segundo BARROS (1997), com o advento da informática, que obteve uma maior popularização com a abertura de reserva de mercado e a conseqüente queda de preços desse tipo de equipamento, as empresas naturalmente se informatizaram. No entanto, para que todos os computadores adquiridos possam ser utilizados plenamente, são necessárias cópias de um

mesmo tipo de *software* a ser instalada em cada unidade, de modo que cada um dos equipamentos possa realizar diversos tipos de tarefas. A utilização de somente um sistema operacional para todos os computadores da rede, obrigatório para que o equipamento se torne funcional, em geral, não possibilita que o computador realize muitas funcionalidades adicionais, salvo por sistemas da família Unix e seus clones, que já possuem uma série de aplicativos e *softwares* distribuídos em conjunto com o sistema operacional.

A falta de comunicação entre usuários e os computadores pode ocasionar o desenvolvimento de trabalhos idênticos em paralelo, resultando dessa forma, na perda de tempo e atraso no desenvolvimento de tarefas importantes. A utilização de computadores em rede possibilita que computadores e periféricos trabalhem em conjunto sem a repetição de tarefas já realizadas, uma vez que esses possuem acesso a bases de dados e informações comuns. Possibilita também um melhor aproveitamento de recursos, pois os computadores podem compartilhar o acesso a recursos de *hardware*, como impressoras, dispositivos de armazenamento de dados, entre outros. Desse modo, o fluxo de trabalho torna-se muito mais produtivo, eliminando a ocorrência de realização de tarefas redundantes e economizando-se recursos na compra de equipamentos ou serviços já existentes.

2.5.1 Conectividade

Um dos requisitos necessários para que computadores possam trabalhar em rede é a conectividade entre eles. Conectividade pode ser descrita como a forma ou método como os computadores se interligam. Em geral, são necessários equipamentos de interconexão de rede e um meio físico pelo qual as informações serão transmitidas.

Os equipamentos de interconexão mais populares são os *switches* e os roteadores. Os *switches* possibilitam que computadores pertencentes a uma mesma rede sejam fisicamente interligados via cabo. Esse tipo de equipamento normalmente possui oito, dezesseis ou vinte e quatro portas. O número de computadores que podem ser interligados é equivalente ao número de portas disponíveis nesse tipo de equipamento. Cada porta do *switch* está associada a um computador distinto por meio de um cabo de rede. Cabe ressaltar que essas associações caracterizam um caminho único e exclusivo entre o *switch* e o computador, ao contrário de

dispositivos menos seguros e de baixo rendimento utilizados anteriormente denominados *hubs*.

Os roteadores possibilitam que redes distintas se comuniquem entre si, permitindo assim, que uma matriz se comunique com suas filiais entre inúmeras outras possibilidades.

Os meios físicos mais comuns utilizados atualmente são cabos, fibras óticas e radiofrequência. A diferença entre cada um deles, além obviamente da natureza de cada um, são a velocidade e a eficiência da transmissão de dados e o custo de implementação. Os meios mais comuns são os cabos de par trançado e radiofrequência, pois possuem custo relativamente baixo e rendimento aceitável para ambientes domésticos e comerciais de pequeno e médio porte e redes locais. Porém, as empresas de grande porte ou de grande volume de tráfego de dados e na interconexão entre matriz e filiais, em geral, optam por conexões de fibra ótica, muito mais rápidas, mas com custo muito mais elevado.

Os equipamentos de rede necessários para o pleno funcionamento da solução proposta são os normalmente utilizados em uma rede local padrão: um *switch*, para interconexão dos equipamentos pertencentes a uma mesma rede local e um *gateway*¹², que possibilita a comunicação entre eles.

2.5.2 Protocolo de comunicação

Para que os dispositivos de interconexão de rede possam trafegar dados gerados pelos computadores e transmitidos pelo meio físico escolhido pelo usuário, é necessário que um procedimento sistemático e padronizado seja utilizado em todos os dispositivos envolvidos nesse tipo de comunicação, um protocolo de comunicação.

O protocolo de comunicação mais difundido atualmente é o protocolo *Transmission Control Protocol/Internet Protocol* (TCP/IP). O protocolo IP permite que os computadores ligados em rede sejam endereçados logicamente e consigam se localizar. O protocolo TCP, garante que a comunicação entre esses dispositivos endereçados pelo protocolo IP troquem informações de maneira confiável, garantindo a entrega de todos os dados necessários para

¹² *Gateway*: ou porta de ligação, é uma máquina intermediária geralmente destinada a interligar redes, separar domínios de colisão, ou mesmo traduzir protocolos. Exemplos de *gateway* podem ser os *routers* (ou roteadores) e *firewalls*, já que ambos servem de intermediários entre o utilizador e a rede. Um *proxy* também pode ser interpretado como um *gateway* (embora em outro nível, aquele da camada em que opera), já que serve de intermediário também.

que a comunicação ocorra. Caso algum dado não seja recebido pelo computador de destino, o protocolo TCP o envia novamente. Isso é possível devido a um mecanismo de sinalização de entrega de pacotes estabelecido pelo protocolo.

A abrangência da comunicação provida pelo protocolo TCP/IP depende das necessidades de seus utilizadores. É possível desde o fornecimento de um número limitado de serviços em abrangência local, uma empresa por exemplo, até a disponibilidade de uma grande gama de funcionalidades por meio da interconexão de equipamentos de infra-estrutura de redes do mundo inteiro, denominados respectivamente de intranet e internet. Em geral, aplicações desenvolvidas para um tipo de ambiente funcionam sem a necessidade de adaptações em outro de maior ou menor abrangência. O usuário deve atentar somente para implicações de segurança que esse tipo de comunicação acarreta.

CAPÍTULO 3. DESENVOLVIMENTO

Este capítulo apresenta a arquitetura proposta para o sistema de registro eletrônico de ponto com gerenciamento remoto. É apresentada, ainda, a arquitetura lógica e o protótipo desenvolvido para validar os componentes previstos nesse trabalho, as interfaces gráficas e o registro dos testes executados.

3.1 Arquitetura física proposta

Os principais elementos sistêmicos presentes na arquitetura proposta neste trabalho são mostrados na Figura 1.

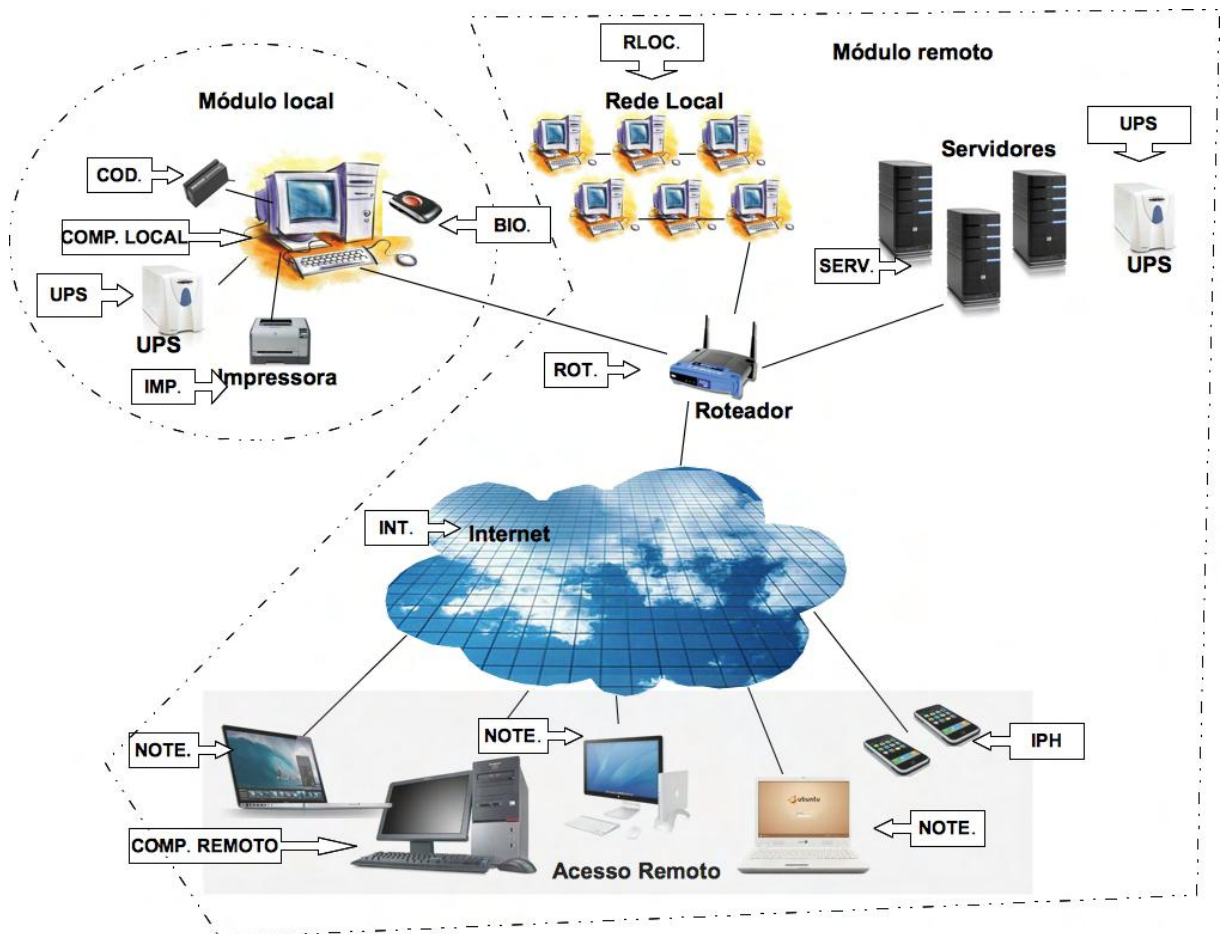


Figura 1: Componentes da arquitetura proposta

Na Figura 1 estão previstos módulos para atender as especificidades regulamentadas pela portaria número 1510 do Ministério do Trabalho e Emprego (MTE), sendo que esses são baseados em *software* livre e código aberto. As interfaces previstas na mencionada arquitetura possibilitam que o usuário utilize e acesse os recursos do sistema por diversos meios. A portaria 1510 do MTE exige somente a utilização de um registrador eletrônico de ponto (COMP.LOCAL) que é utilizado exclusivamente para esse fim, com capacidade de funcionamento ininterrupto por um período mínimo de mil quatrocentos e quarenta horas na ausência de energia elétrica, recurso esse fornecido na arquitetura proposta por meio de uma unidade ininterrupta de energia elétrica (UPS). O registrador eletrônico de ponto deve estar conectado diretamente a uma impressora interna ou externa (IMP.) a fim de possibilitar a impressão dos comprovantes individuais de registro de ponto. O módulo local é representado pelo registrador eletrônico de ponto, pela impressora e pela fonte de energia secundária (UPS).

O módulo remoto complementa as funcionalidades do registrador eletrônico de ponto que estão relacionadas com o acesso remoto, o qual é denominado “gerenciamento remoto”. Esse módulo é composto por outros componentes que foram previstos com a finalidade de permitir o acesso remoto aos recursos do sistema proposto. Esse recurso possibilita o acesso às informações cadastrais, referentes aos registros armazenados em uma base de dados instalada em um servidor específico (SERV.) na rede local. O meio para acessar essas funcionalidades pode ser de modo local, via rede local (RLOC.), ou ainda por meio da rede mundial de computadores (internet – INT.). Esse último com a utilização de programas que permitem aos usuários a interagirem com documentos virtuais por meio de navegadores *web* (*browser*), disponíveis em computadores (COMP.REMOTO), notebooks (NOTE.), celulares (IPH.) ou qualquer outro tipo de equipamento eletrônico com possibilidades de acesso à internet.

O roteador (ROT.) presente na arquitetura tem por finalidade permitir que as informações sejam acessadas por meio dos padrões de *hardware* e *software* definidos para utilizar a rede mundial de computadores (INT.).

Em sistemas de registro de presença é comum a utilização de dispositivos biométricos¹³. Atualmente, este termo também é associado à medida de características físicas ou comportamentais das pessoas como forma de identificá-las unicamente. É usada na

¹³ *Biometria*: [bio (vida) + metria (medida)]. É o estudo estatístico das características físicas ou comportamentais dos seres vivos

identificação criminal, no controle de acesso a áreas restritas, entre outras aplicações. Os sistemas chamados biométricos podem basear o seu funcionamento em características de diversas partes do corpo humano, como por exemplo: os olhos, a palma da mão, as digitais do dedo, a retina ou íris dos olhos. A premissa em que se fundamentam é a de que cada indivíduo é único e possui características físicas e de comportamento distintas.

A arquitetura do protótipo elaborada para esse trabalho não prevê a utilização de mecanismos biométricos como tipo de identificação, pois este tipo de mecanismo não é exigido pela portaria 1510 do MTE. Os mecanismos de autenticação utilizados neste trabalho são triviais e sobretudo, funcionais, pois são baseados na utilização de usuário e senha, como a grande maioria dos serviços oferecidos atualmente. É importante destacar que a utilização e o suporte aos mecanismos de identificação biométrica podem ser incorporados a aplicação apresentada nesse trabalho, bastando para tanto, efetuar a adequação do módulo de registro de usuário. O mesmo conceito poderá ser aplicado a utilização de crachás com códigos de barras, cujos problemas de uso não se limitam somente ao equipamento de leitura, mas também a fragilidade e o desgaste dos códigos de barras impressos que também podem impossibilitar o registro de ponto. Apesar desses tipos de equipamentos não serem exigidos pela portaria 1510 do MTE, a arquitetura proposta nesse trabalho elucidada em qual contexto esses dispositivos estariam inseridos se fossem necessários incluí-los. Para tanto, foram especificados um leitor biométrico de impressões digitais (BIO.) e um leitor de código de barras (COD.), ambos presentes no módulo local.

3.2 Requisitos de *hardware* e *software*

Segundo MTE (2009), os requisitos de *hardware* e *software* necessários para funcionamento e comercialização dos registradores eletrônicos de ponto são listados na Tabela 1 e Tabela 2. Nessas tabelas, os requisitos de *software* indicados com a letra “X” na coluna apto são atendidos pelo sistema desenvolvido neste trabalho.

Tabela 1: Requisitos exigidos pela Portaria 1510 – primeira fase

Item	Descrição de requisitos	Tipo	Apto
1	O sistema de registro eletrônico de ponto deve registrar fielmente as marcações efetuadas.	Funcional	X
2	Para a utilização de sistema de registro eletrônico de ponto é obrigatório o uso do registrador eletrônico de ponto (REP) no local de prestação de serviço.	Não funcional	-
3	O REP deverá apresentar relógio interno de tempo real com precisão mínima de um minuto por ano.	Não funcional	-
4	O REP deverá apresentar mostrador do relógio de tempo real contendo hora, minutos e segundos.	Não funcional	-
5	O REP deverá dispor de mecanismo impressor em bobina de papel, integrado e de uso exclusivo do equipamento.	Não funcional	-
6	O REP deverá dispor de um meio de armazenamento permanente, denominado Memória de Registro de Ponto (MRP), onde ficarão armazenados os dados necessários à operação de REP.	Não funcional	X
7	Na memória permanente (MRP) deverá ser gravado as operações de inclusão ou alteração das informações do empregador na MT.	Funcional	X
8	Na memória permanente (MRP) deverá ser gravado as marcações de ponto.	Funcional	X

Tabela 2: Requisitos exigidos pela Portaria 1510 – segunda fase

Item	Descrição requisitos	Tipo	Apto
1	Na memória permanente (MRP) deverá ser gravado os ajustes do relógio interno.	Funcional	X
2	Na memória permanente (MRP) deverá ser gravado as operações de inclusão, alteração e exclusão de dados do empregado na MT.	Funcional	X
3	O REP deverá dispor de um meio de armazenamento, denominado Memória de Trabalho (MT), onde ficarão armazenados os dados necessários à operação de REP.	Não funcional	X
4	Na memória de trabalho (MT) deverá ser armazenado os dados referentes ao empregado.	Funcional	X
5	Na memória de trabalho (MT) deverá ser armazenado os dados referentes ao empregador.	Funcional	X
6	O REP deverá ter uma porta padrão USB externa, denominada porta fiscal.	Não funcional	-
7	O REP não deverá depender de qualquer conexão com outro equipamento externo para a marcação do ponto.	Não funcional	-
8	A marcação de ponto ficará interrompida quando for feita qualquer operação que exija a comunicação com o REP com qualquer outro equipamento, seja para carga ou leitura de dados.	Não funcional	-
9	A marcação do ponto deverá ocorrer na seguinte ordem: receber diretamente a identificação do trabalhador, sem interposição de outro equipamento; obter a hora do relógio de tempo real; registrar a marcação de ponto na MRP; imprimir o comprovante do trabalhador.	Funcional	X
10	Geração do arquivo-fonte de dados (AFD) a partir dos dados armazenados na MRP.	Funcional	X
11	Gravação do AFD em dispositivo externo de memória, por meio da porta fiscal.	Não funcional	-
12	Emissão da relação instantânea de marcações com as marcações efetuadas nas vinte e quatro horas precedentes.	Funcional	X
13	Emissão do comprovante de registro de ponto do trabalhador para o empregado acompanhar, a cada marcação, o controle de sua jornada de trabalho.	Funcional	X

3.3 Arquitetura lógica

Nesta seção é apresentada a arquitetura de lógica, contendo o comparativo dos requisitos exigidos e os atendidos, os casos de uso gráfico e textuais e o diagrama de classes que foram identificados e analisados para a implementação do sistema, sendo validados por meio de testes práticos.

3.3.1 Requisitos exigidos pela portaria e implementados pelo trabalho

Na subseção 3.2, nas Tabelas 1 e 2, são listados os 21 requisitos identificados durante o levantamento de requisitos exigidos pela portaria no. 1510, dos quais 14 são implementados por esse trabalho.

Na Tabela 1, o requisito item 1 é implementado pelo módulo de acesso local, por meio do qual o usuário ao efetuar o registro eletrônico de ponto, não tem acesso algum às informações registradas na memória permanente (MRP). O mesmo ocorre para os usuários que possuem acesso ao módulo remoto, também não há disponível uma interface gráfica que possibilite alteração dos dados registrados nos devidos meios de armazenamento, memória de trabalho (MT) e permanente (MRP).

Os requisitos não funcionais: item 6 da Tabela 1 e item 3 da Tabela 2 são atendidos por meio da utilização de um banco de dados, contendo as tabelas MRP e MT, respectivamente.

Os requisitos funcionais: item 7 e 8 da Tabela 1 e item 1, 2, 4 e 5 da Tabela 2 são atendidos por *scripts* específicos de inserção na tabela MRP.

O requisito funcional: item 9 da Tabela 2 é atendido pelo *script* de registro do ponto executado a partir do módulo de acesso local.

O requisito funcional: item 10 da Tabela 2 é atendido pela interface gráfica Arquivos de Fiscalização por meio de acesso remoto no menu de Cadastros.

O requisito funcional: item 12 da Tabela 2 é atendido pela interface gráfica Espelho de Ponto por meio de acesso remoto no menu de Relatórios.

O requisito funcional: item 13 da Tabela 2 é atendido pelo *script* de registro do ponto executado a partir do módulo de acesso local.

3.3.2 Casos de uso gráficos

As principais funcionalidades identificadas e implementadas são representadas por meio do diagrama de caso de uso local e remoto do sistema, conforme apresentado na Figura 2 e Figura 3, respectivamente.

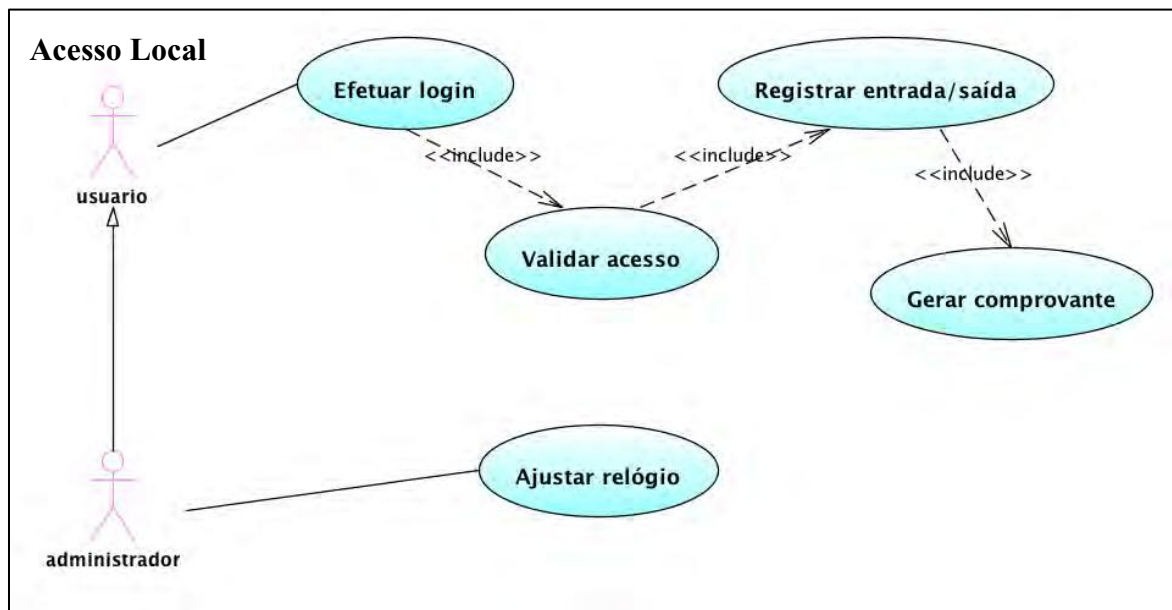


Figura 2: Caso de Uso Gráfico para o acesso local

Na Figura 3 é representado as funcionalidades identificadas para o acesso remoto.

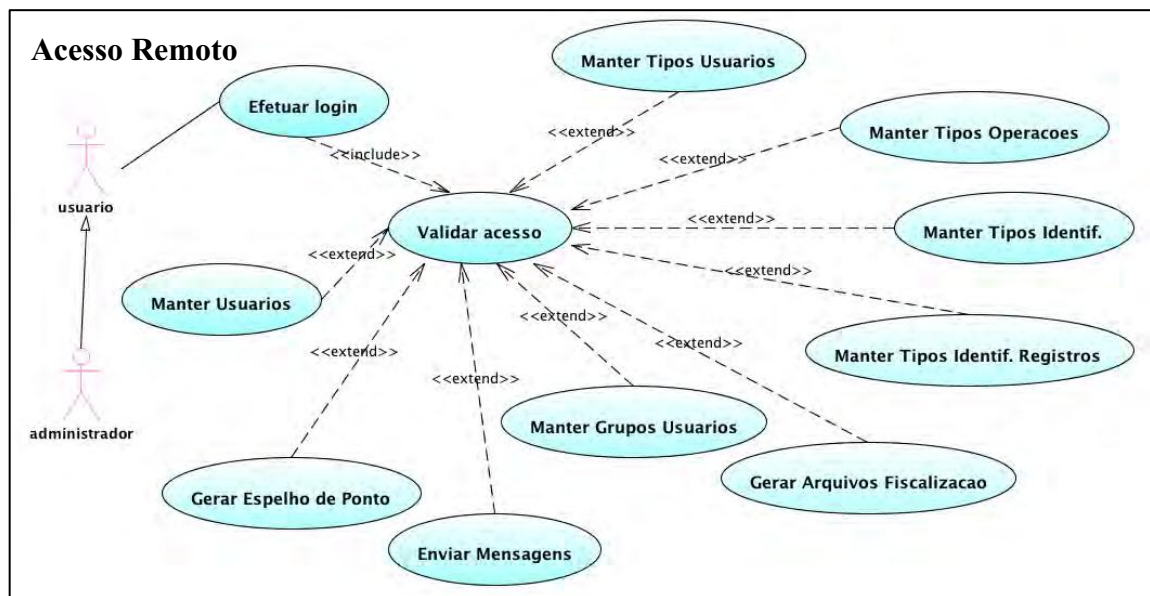


Figura 3: Caso de Uso Gráfico para o acesso remoto

3.3.3 Casos de uso textuais

Baseado no levantamento de requisitos de *software*, apresentados na subseção 3.3.1 são identificados os casos de uso (*Use Case - UC*) listados nessa subseção. Os casos de uso textuais formalizam as funcionalidades identificadas e descrevem o comportamento do sistema sob a visão dos usuários finais, analistas e equipe de testes.

3.3.3.1 Casos de uso do módulo local

Nessa subseção são descritos os casos de uso necessários para o desenvolvimento do módulo local.

3.3.3.1.1 UC Efetuar *login*

Este caso de uso tem por objetivo permitir o acesso ao módulo local de forma que o usuário possa efetuar o registro de ponto. A pré-condição para que este caso de uso se inicie é que o usuário deve estar cadastrado no sistema, na memória de trabalho (MT).

Para se efetuar o *login*, o usuário deve informar, obrigatoriamente, a sua identificação de acesso. Em seguida, a sua senha de identificação. O caso de uso UC Validar acesso é acionado conforme apresentado na subseção 3.3.3.1.2. Esse caso de uso é finalizado.

3.3.3.1.2 UC Validar acesso

Este caso de uso tem por objetivo validar o acesso ao módulo local permitindo ao usuário efetuar o registro de ponto. As informações de identificação e senha recebidos são utilizados para validar o acesso por meio de comparação com os dados do usuário gravados na memória de trabalho (MT). Na situação em que esse é validado, o caso de uso UC Registrar entrada/saída é acionado, conforme apresentado na subseção 3.3.3.1.3. Na situação em que esse está incorreto, seja a identificação ou senha, é exibida a interface gráfica de erro. Esse caso de uso é finalizado.

3.3.3.1.3 UC Registrar entrada/saída

Este caso de uso tem por objetivo registrar no banco de dados o registro de ponto do usuário na hora local. Esse caso de uso se inicia, no momento em que recebe as informações necessárias para registrar a entrada/saída do usuário na memória permanente (MRP). Os dados pertinentes ao usuário, data e hora do registro são gravados na memória permanente.

Em seguida, o caso de uso UC Gerar comprovante é acionado, conforme apresentado na subseção 3.3.3.1.4. Esse caso de uso é finalizado.

3.3.3.1.4 UC Gerar comprovante

Este caso de uso tem por objetivo gerar e emitir o comprovante de registro de ponto ao usuário. Esse caso de uso se inicia no momento em que é acionado, buscando as informações salvas no banco de dados, especificamente, na memória permanente (MRP) para impressão do comprovante. Em seguida, é exibida a interface gráfica de sucesso ao registrar o ponto, retornando ao caso de uso UC Efetuar *login* para que o próximo usuário possa efetuar o registro, conforme apresentado na subseção 3.3.3.1.1. Esse caso de uso é finalizado.

3.3.3.1.5 UC Ajustar relógio

Este caso de uso tem por objetivo registrar as alterações que possam ocorrer no ajuste do relógio utilizado para o registro eletrônico do ponto. Esse caso de uso se inicia no momento em que há necessidade de alterar o horário do relógio e o administrador informa o novo horário, e após confirmar, os horários antigo e o novo são registrados no banco de dados, na memória permanente (MRP). Esse caso de uso é finalizado.

3.3.3.2 Casos de uso do módulo remoto

Nessa subseção são descritos os casos de uso necessários para o desenvolvimento do módulo remoto.

3.3.3.2.1 UC Efetuar *login*

Este caso de uso tem por objetivo permitir o acesso ao módulo remoto de forma que o usuário possa acessar o menu de opções de cadastros, consultas e relatórios. A pré-condição para que este caso de uso se inicie é que o usuário deve estar cadastrado no sistema, na memória de trabalho (MT).

Para se efetuar o *login*, o usuário deve informar, obrigatoriamente, a sua identificação de acesso. Em seguida, a sua senha de identificação. O caso de uso UC Validar acesso é acionado, conforme apresentado na subseção 3.3.3.2.2. Esse caso de uso é finalizado.

3.3.3.2.2 UC Validar acesso

Este caso de uso tem por objetivo validar o acesso ao módulo remoto permitindo ao usuário acessar o menu de opções. As informações de identificação e senha recebidos são utilizados para validar o acesso por meio de comparação com os dados do usuário gravados na memória de trabalho (MT). Na hipótese em que este é validado, é exibido a interface gráfica com o menu de opções. Mediante a escolha de uma opção dos menus disponíveis, o usuário é remetido a um dos casos de uso listados nas subseções 3.3.3.2.3 até 3.3.3.2.11. Na situação em que o acesso não é validado, é exibida uma mensagem de erro, sendo que o usuário deverá tentar novamente o acesso. Esse caso de uso é finalizado.

3.3.3.2.3 UC Manter tipos de usuários

Este caso de uso tem por objetivo manter¹⁴ os tipos de usuários utilizados no acesso local, bem como no remoto. O usuário seleciona a opção de cadastro no menu de opções. Escolhe a interface gráfica referente a tipos de usuários. É exibida a interface contendo os

¹⁴ manter: significa que é aplicado as 4 operações básicas utilizadas em banco de dados - CRUD (*create, read, update e delete*).

campos necessários para inserir ou atualizar os tipos de usuários. Para a situação de inserção, basta informar a descrição do tipo de usuário. É exibida uma mensagem de confirmação de inserção com sucesso. Para a situação de atualização, deve-se informar a identificação do tipo de usuário a ser atualizado. É exibida a interface gráfica contendo todos os dados do tipo de usuário selecionado.

O usuário seleciona a opção de consulta no menu de opções. Escolhe a interface gráfica referente a tipos de usuário. É exibida a interface para a situação de consulta, onde o usuário deve informar a faixa inicial e final da pesquisa. Para a situação de exclusão, deve-se informar a identificação do tipo de usuário a ser excluído. Sendo permitido somente se o registro ainda não foi relacionado. Essa integridade é mantida pelo sistema de gerenciamento de banco de dados. Fim do caso de uso.

3.3.3.2.4 UC Manter tipos de operações

Este caso de uso tem por objetivo manter os tipos de operações utilizadas no registro das informações na memória permanente. O usuário seleciona a opção de cadastro no menu de opções. Escolhe a interface gráfica referente a tipos de operações. É exibida a interface contendo os campos necessários para inserir ou atualizar os tipos de operações. Para a situação de inserção, informar a descrição do tipo de operação. É exibida uma mensagem de confirmação de inserção com sucesso. Para a situação de atualização, deve-se informar a identificação do tipo de operação a ser atualizada. É exibida a interface gráfica contendo todos os dados do tipo de operação selecionada.

O usuário seleciona a opção de consulta no menu de opções. Escolhe a interface gráfica referente a tipos de operações. É exibida a interface para a situação de consulta, onde o usuário deve informar o range inicial e final da pesquisa. Para a situação de exclusão, deve-se informar a identificação do tipo de operação a ser excluída. Sendo permitido somente se o registro ainda não foi relacionado. Essa integridade é mantida pelo sistema de gerenciamento de banco de dados. Fim do caso de uso.

3.3.3.2.5 UC Manter tipos de identificação

Este caso de uso tem por objetivo manter os tipos de identificação utilizados no registro das informações na memória de trabalho e permanente para o módulo de acesso local e remoto. O usuário seleciona a opção de cadastro no menu de opções. Escolhe a interface gráfica referente a tipos de identificação. É exibida a interface contendo os campos necessários para inserir ou atualizar os tipos de identificação. Para a situação de inserção, basta informar a descrição do tipo de identificação. É exibida uma mensagem de confirmação de inserção com sucesso. Para a situação de atualização, deve-se informar o identificador do tipo de identificação a ser atualizada. É exibida a interface gráfica contendo todos os dados do tipo de identificação selecionada.

O usuário seleciona a opção de consulta no menu de opções. Escolhe a interface gráfica referente a tipos de identificação. É exibida a interface para a situação de consulta, onde o usuário deve informar a faixa inicial e final da pesquisa. Para a situação de exclusão, deve-se informar o identificador do tipo de identificação a ser excluída. Sendo permitido somente se o registro ainda não foi relacionado. Essa integridade é mantida pelo sistema de gerenciamento de banco de dados. Fim do caso de uso.

3.3.3.2.6 UC Manter tipos de identificação de registros

Este caso de uso tem por objetivo manter os tipos de identificação de registros utilizados no registro das informações na memória permanente para o módulo de acesso remoto. O usuário seleciona a opção de cadastro no menu de opções. Escolhe a interface gráfica referente a tipos de identificação de registro. É exibida a interface contendo os campos necessários para inserir ou atualizar os tipos de identificação de registro. Para a situação de inserção, basta informar a descrição do tipo de identificação de registro. É exibida uma mensagem de confirmação de inserção com sucesso. Para a situação de atualização, deve-se informar o identificador do tipo de identificação de registro a ser atualizada. É exibida a interface gráfica contendo todos os dados do tipo de identificação de registro selecionada.

O usuário seleciona a opção de consulta no menu de opções. Escolhe a interface gráfica referente a tipos de identificação de registro. É exibida a interface para a situação de consulta, onde o usuário deve informar a faixa inicial e final da pesquisa. Para a situação de exclusão, deve-se informar o identificador do tipo de identificação de registro a ser excluída. Sendo permitido somente se o registro ainda não foi relacionado. Essa integridade é mantida pelo sistema de gerenciamento de banco de dados. Fim do caso de uso.

3.3.3.2.7 UC Manter grupos de usuários

Este caso de uso tem por objetivo manter os grupos de usuários utilizados no registro das informações do usuário para o módulo de acesso remoto. O usuário seleciona a opção de cadastro no menu de opções. Escolhe a interface gráfica referente a grupos de usuários. É exibida a interface contendo os campos necessários para inserir ou atualizar os grupos de usuários. Para a situação de inserção, basta informar a descrição do grupos de usuários. É exibida uma mensagem de confirmação de inserção com sucesso. Para a situação de atualização, deve-se informar o identificador do grupos de usuários a ser atualizado. É exibida a interface gráfica contendo todos os dados do grupo de usuários selecionada.

O usuário seleciona a opção de consulta no menu de opções. Escolhe a interface gráfica referente a grupos de usuários. É exibida a interface para a situação de consulta, onde o usuário deve informar a faixa inicial e final da pesquisa. Para a situação de exclusão, deve-se informar o identificador do grupos de usuários a serem excluídos. Sendo permitido somente se o registro ainda não foi relacionado. Essa integridade é mantida pelo sistema de gerenciamento de banco de dados. Fim do caso de uso.

3.3.3.2.8 UC Manter usuários

Este caso de uso tem por objetivo manter os usuários utilizados no registro das informações do usuário para o módulo de acesso local e remoto. O usuário seleciona a opção de cadastro no menu de opções. Escolhe a interface gráfica referente a usuários. É exibida a

interface contendo os campos necessários para inserir ou atualizar os usuários. Para a situação de inserção, informar os dados pertinentes ao cadastro do usuário. É exibida uma mensagem de confirmação de inserção com sucesso. Para a situação de atualização, deve-se informar o identificador do usuário a ser atualizado. É exibida a interface gráfica contendo todos os dados do usuário selecionado.

O usuário seleciona a opção de consulta no menu de opções. Escolhe a interface gráfica referente aos usuários. É exibida a interface para a situação de consulta, onde o usuário deve informar a faixa inicial e final da pesquisa. Para a situação de exclusão, deve-se informar o identificador do usuário a ser excluído. Sendo permitido somente se o registro ainda não foi relacionado. Essa integridade é mantida pelo sistema de gerenciamento de banco de dados. Fim do caso de uso.

3.3.3.2.9 UC Enviar mensagem

Este caso de uso tem por objetivo permitir aos usuários enviar mensagens de *e-mail*¹⁵ para os usuários do sistema, bem como usuários externos. O usuário seleciona a opção de cadastro no menu de opções. Escolhe a interface gráfica referente a mensagens gerais. É exibida a interface para inserir as informações desejadas e informar os *e-mails* destinatários e enviar. O caso de uso é finalizado.

3.3.3.2.10 UC Gerar arquivos da fiscalização

Este caso de uso tem por objetivo gerar os arquivos exigidos pela portaria. O usuário seleciona a opção de cadastro no menu de opções. Escolhe a interface gráfica referente a arquivos da fiscalização. É exibida a interface para selecionar as datas iniciais e finais para a geração dos arquivos. Informa também o identificador do empregador. Solicita a geração e o

¹⁵ *Email*: correio eletrônico. É um meio que permite compor, enviar e receber mensagens através de sistemas eletrônicos de comunicação. O termo *e-mail* é aplicado tanto aos sistemas que utilizam a *Internet* e são baseados no protocolo SMTP, como aqueles sistemas conhecidos como *intranets*, que permitem a troca de mensagens dentro de uma empresa ou organização e são, normalmente, baseados em protocolos proprietários.

caso de uso é finalizado. Os arquivos ficam disponíveis para serem salvos pela porta padrão USB externa.

3.3.3.2.11 UC Gerar espelho de ponto

Este caso de uso tem por objetivo gerar o espelho¹⁶ de ponto. O usuário seleciona a opção de relatórios no menu de opções. Escolhe a interface gráfica referente a espelho de ponto. A pré-condição para que este caso de uso se inicie é que o usuário deve estar cadastrado no sistema. É exibida a interface para selecionar as datas iniciais e finais para a geração do espelho de ponto. Solicita a geração, sendo direcionado para a impressão *default*. O caso de uso é finalizado.

¹⁶ Espelho de ponto: relatório gerado com *layout* específico, o qual contém informações de entrada e saída dos usuários dos sistemas de registros de ponto.

3.3.4 Diagrama de classe

Na Figura 4 é representado o diagrama de classe conceitual simplificado do sistema proposto. Contém as principais classes e suas respectivas multiplicidades identificadas nessa etapa. Os métodos identificados são descritos no fluxograma analítico de acesso local e remoto.

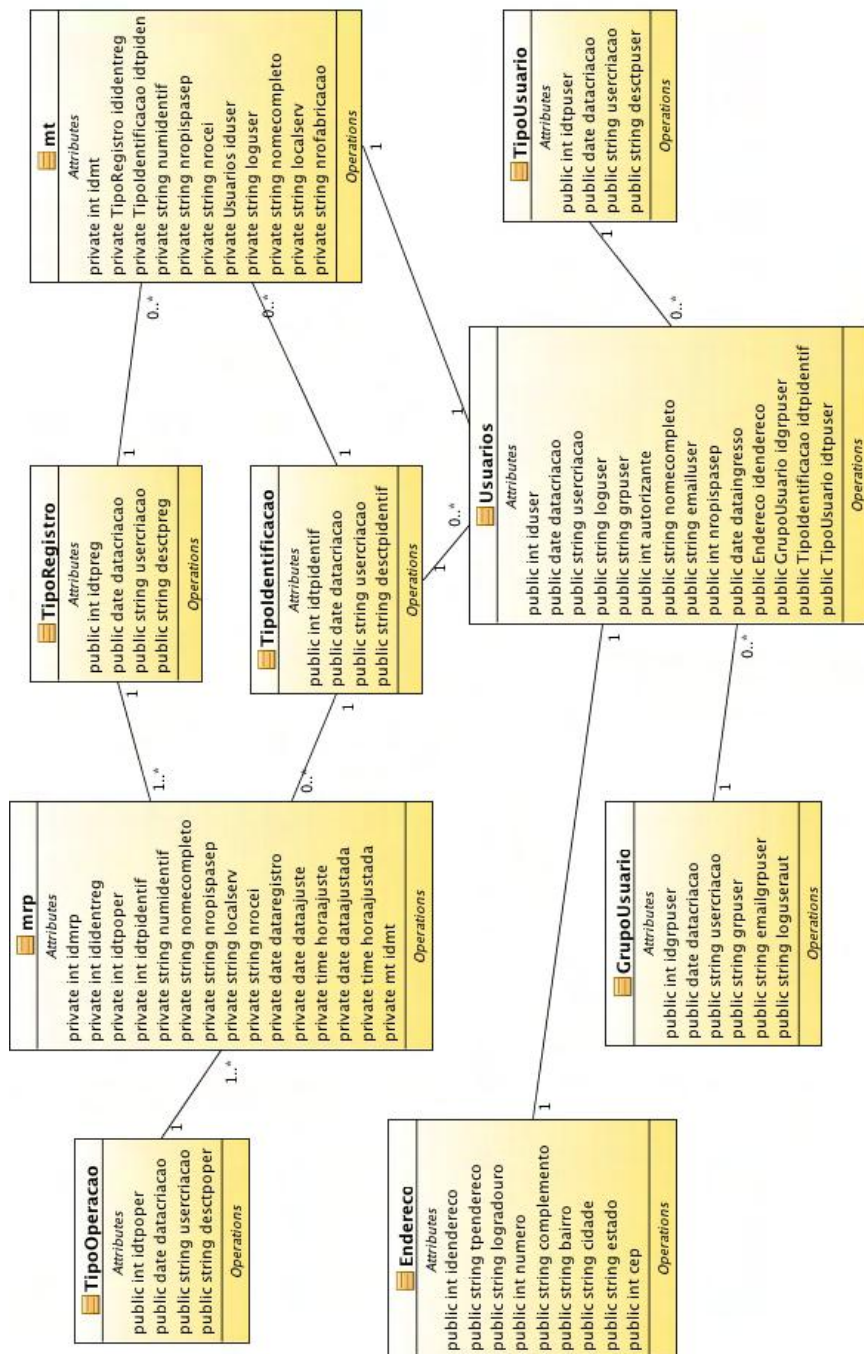


Figura 4: Diagrama de Classes

3.4 Protótipo

Nesta seção são apresentadas a arquitetura de *hardware* e *software* que foram utilizados nos testes práticos para validação do proposto sistema de registro eletrônico de ponto, conforme apresentado na Figura 5. A representação do módulo local utilizado para efetuar o registro eletrônico de ponto é realizada pelo item 1 e a representação do módulo de acesso remoto utilizado para gerenciamento pelo item 2. A conectividade de rede é proporcionada pelo uso de roteador, representado pelo item 3. A impressão do comprovante de registro de ponto é feita por meio do item 4.

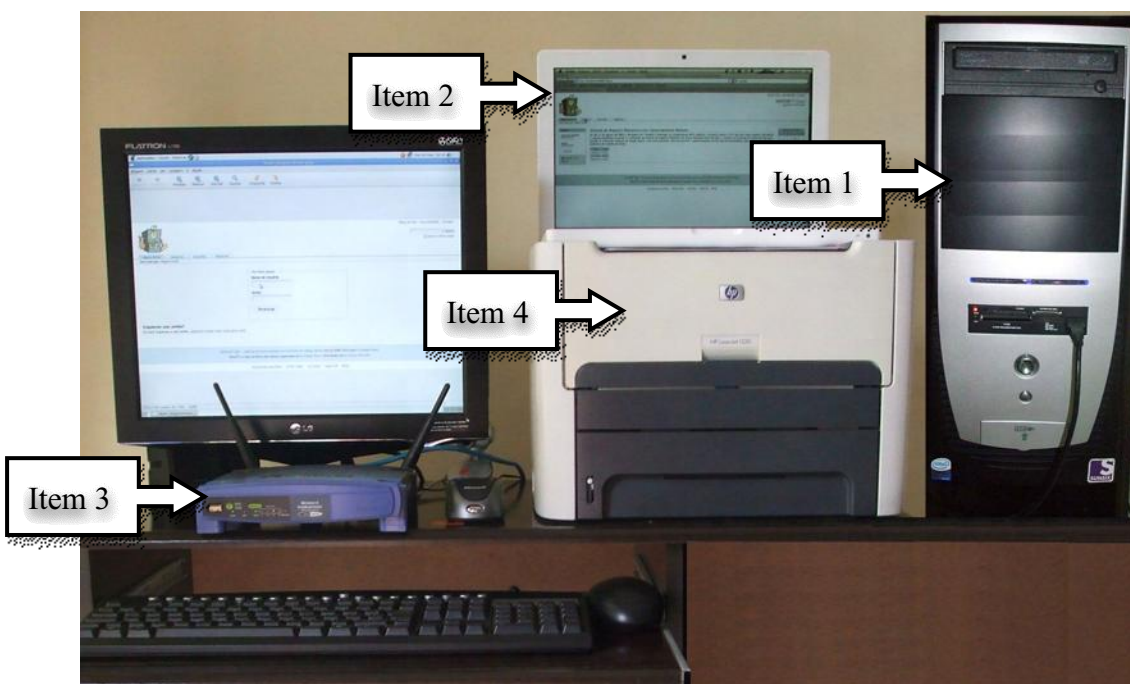


Figura 5: Equipamentos utilizados nos testes práticos

3.4.1 Arquitetura do protótipo

Neste protótipo foram realizados os testes para validar os principais blocos da arquitetura proposta neste trabalho, os quais estão representados na Figura 6. A arquitetura

utilizada para a realização dos testes práticos de registro eletrônico de ponto, é baseada na arquitetura completa apresentada na Figura 1, no entanto, é composta somente por um computador servidor, representado pelo módulo remoto e por um computador local e uma impressora, representados pelo módulo local na Figura 6.

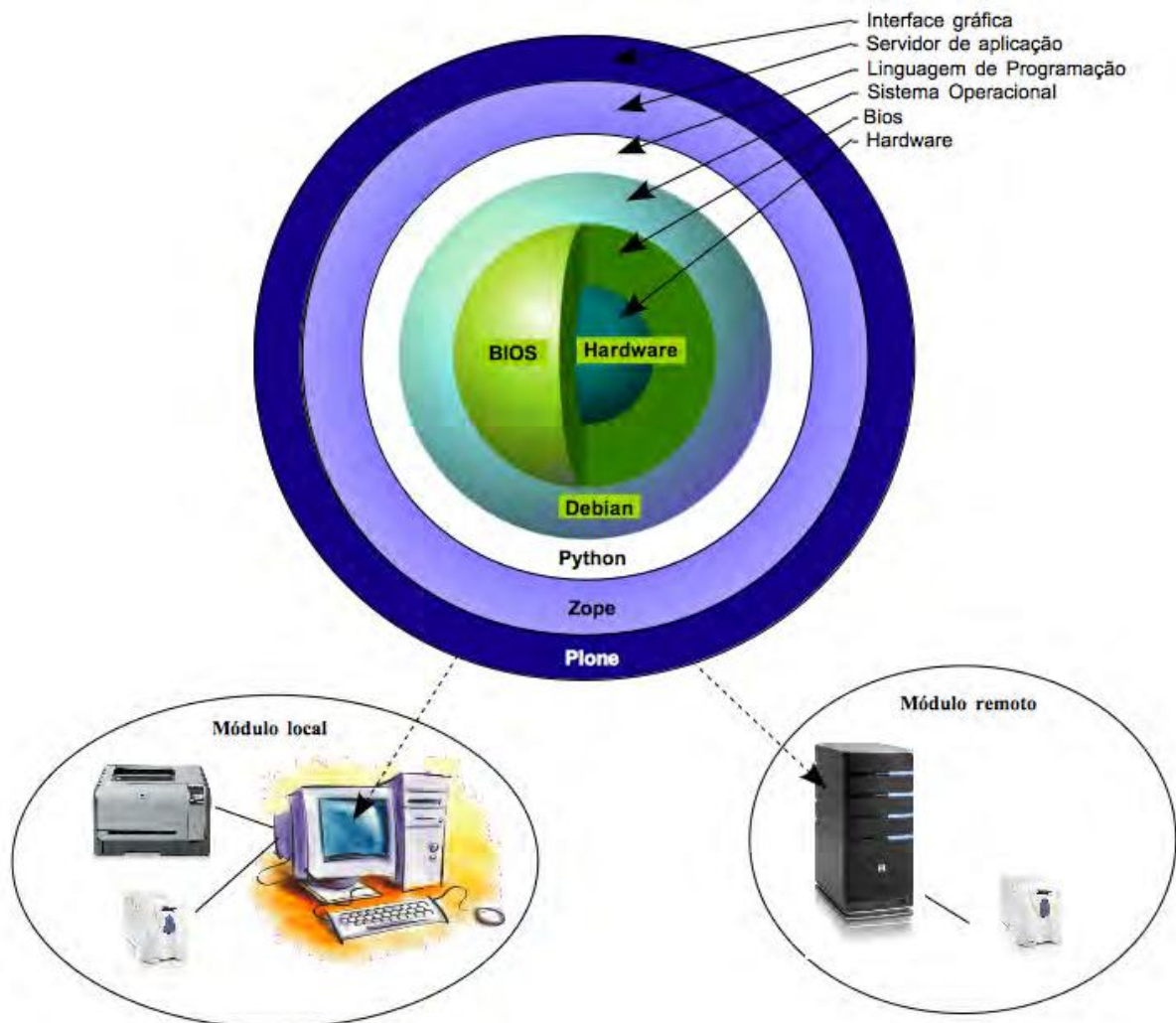


Figura 6: Arquitetura do protótipo

Na Figura 6, o computador servidor utilizado possui os seguintes *softwares* instalados para o funcionamento do sistema desenvolvido, a saber: sistema operacional GNU/Linux – distribuição Debian Lenny versão 5.06 (DEBIAN, 2010) compilado para a plataforma Intel

x86 de 32 bits, linguagem de programação python (PYTHON, 2010) e o servidor de aplicação Zope (ZOPE, 2010) com o *framework* web Plone (PLONE, 2010). Além disso, no módulo local, o computador local foi configurado com o sistema operacional principal que reproduz os requisitos de *hardware* exigidos pela portaria 1510 para a produção do dispositivo de registro eletrônico de ponto.

Os *softwares* utilizados são os mesmos do computador servidor. Cada uma das máquinas possui um endereço de rede distinto e o ambiente provê conectividade de rede entre elas pela utilização de roteadores ou outro tipo de equipamento de infra-estrutura de redes. A impressora (IMP.) utilizada tem como única finalidade a impressão dos comprovantes de registro de ponto, conforme previsto na portaria do MTE.

3.4.2 Programa de gerenciamento

O sistema proposto é baseado na linguagem de programação python. A interface gráfica disponível nos navegadores *web*, local ou remotamente, é provida pelo *framework* web plone, que por sua vez, é instalado no servidor de aplicações Zope. Ambos os *softwares* são desenvolvidos em linguagem python, compatíveis entre si. Os *softwares* utilizados nesta arquitetura de protótipo (python, zope, plone e mysql) são multiplataforma, e sua utilização garante a compatibilidade com uma grande variedade de sistemas operacionais e arquiteturas de computadores. Esse fator traz como consequência direta a capacidade de expansão nas possíveis soluções de *hardware* e *software* suportadas.

No módulo local, a interface com o usuário é baseada em um navegador web. A impressão dos comprovantes de registro de entrada e saída é gerada em Formato de Documento para Impressão (*Printable Document Format* - PDF) diretamente para a impressora.

As principais funcionalidades exigidas pela portaria número 1510 estão presentes na solução proposta e são realizadas com a execução de *scripts* python associados a ações específicas, previstas previamente, disparadas pela interação do usuário com os formulários. Algumas funcionalidades também são disparadas por tarefas agendadas do sistema, tais como, sincronização do banco de dados da aplicação com a memória de trabalho (MT) disponível no registrador eletrônico de ponto.

O algoritmo do programa de gerenciamento de acesso local neste trabalho executa, principalmente, os seguintes passos:

Passo 1 – Iniciar acesso a sistema local?

Não, verificar algoritmo de acesso remoto.

Sim, acessar o sistema de registro eletrônico de ponto.

Executar Passo 2.

Passo 2 – Informar usuário e senha.

Passo 3 – Acesso válido?

Não, retorna ao Passo 2.

Sim, validar usuário na memória de trabalho (MT).

Executar Passo 3.

Passo 3 – Usuário válido na memória de trabalho?

Não, retorna ao Passo 2.

Sim, efetuar o registro eletrônico de ponto, gravando na memória de registro de ponto (MRP). Emitir o comprovante de entrada e saída.

Executar Passo 4.

Passo 4 – Efetuar o *logout*.

Finaliza a operação.

Executar Passo 1.

O fluxograma analítico que representa essas operações é representado na Figura 7. A operação em modo de acesso local, tem por objetivo efetuar o registro eletrônico de ponto. O usuário informa o seu número de identificação e senha respectivamente. O programa de gerenciamento local valida essas informações na memória de trabalho (MT) do banco de dados da aplicação disponível no registrador eletrônico de ponto, registra o ponto e emite o comprovante de registro. Efetua *logout*, permitindo o registro de ponto do próximo usuário.

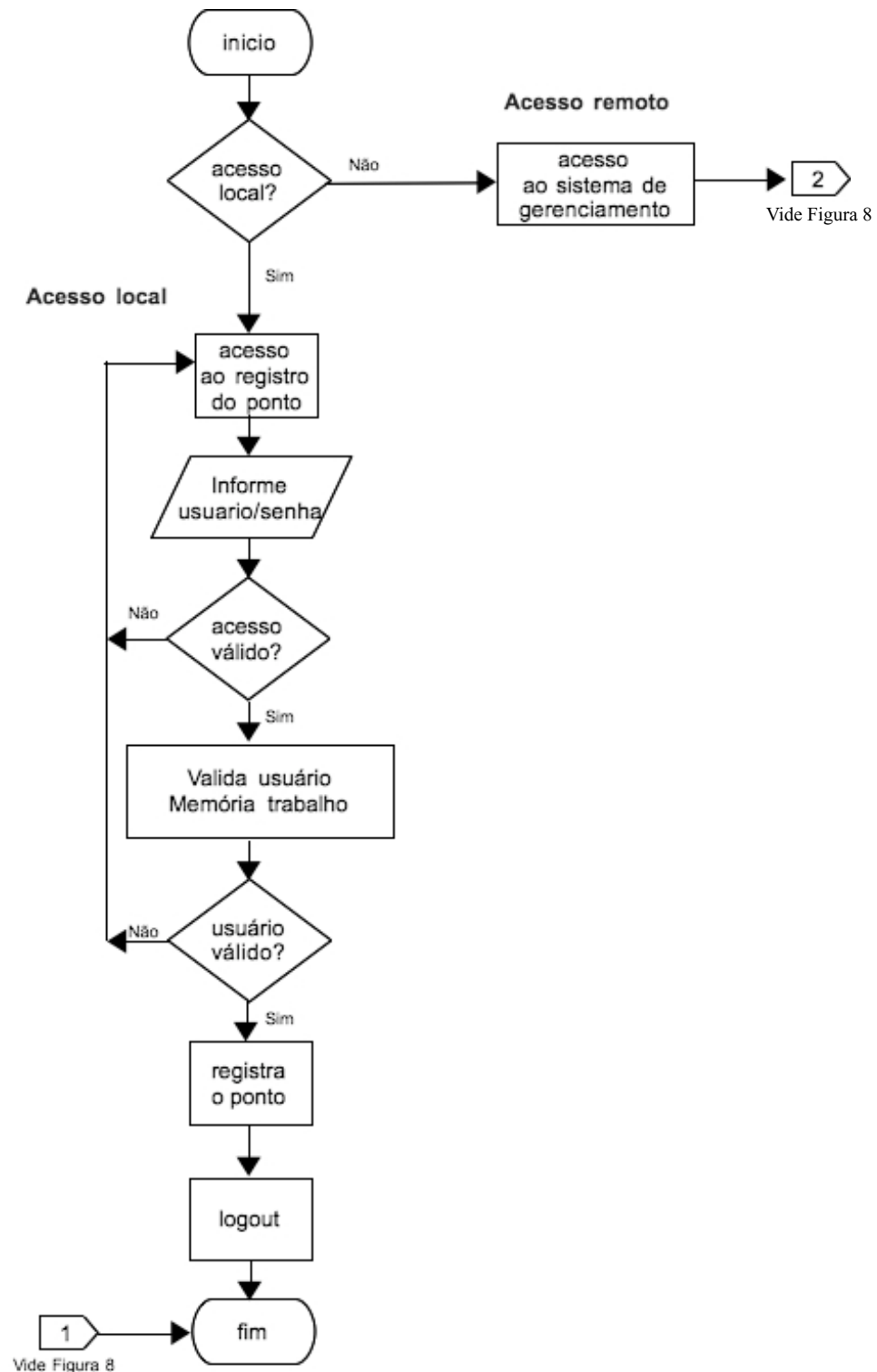


Figura 7: Fluxograma de acesso local

No módulo de gerenciamento remoto estão disponíveis informações referentes aos colaboradores, empregadores, registro de ponto. Esses dados são armazenados em tabelas do tipo Linguagem Estruturada de Pesquisa (*Structured Query Language - SQL*). Para que esses

dados possam ser manipulados, organizados e consultados de maneira eficiente, é necessária a utilização de um Sistema Gerenciador de Banco de Dados (SGBD). Nessa arquitetura do protótipo, optou-se pelo SGBD livre MySQL (MYSQL, 2010). As janelas de cadastro, edição, remoção e consulta de dados foram elaboradas com a utilização de módulos de *software* disponíveis para o *framework web* Plone. Um desses *plugins* adicionais, é o PloneFormGen, utilizado para a elaboração e personalização de formulários. A instalação de um *plugin* denominado MySQLdb também foi necessário, uma vez que o *framework web* Plone não possui suporte a esse SGBD originalmente.

O algoritmo do programa de gerenciamento de acesso remoto neste trabalho executa, principalmente, os seguintes passos:

Passo 1 – Iniciar acesso a sistema local?

Não, acessar o sistema de gerenciamento remoto.

Executar Passo 2.

Sim, verificar algoritmo de acesso local.

Executar Passo 1 do sistema de registro eletrônico de ponto, acesso local.

Passo 2 – Informar usuário e senha.

Passo 3 – Acesso válido?

Não, retorna ao Passo 2.

Sim, exibir página com opções de gerenciamento remoto.

Executar Passo 4.

Passo 4 – Página com opções de menus.

Passo 5 – Opção escolhida é Menu de Cadastros?

Não, executar Passo 6.

Sim, executar Passo 8.

Passo 6 – Opção escolhida é Menu de Consultas?

Não, executar Passo 7.

Sim, executar Passo 9.

Passo 7 – Opção escolhida é Menu de Relatórios?

Não, executar Passo 28.

Sim, executar Passo 10.

Passo 8 – Exibe opções do Menu de Cadastros.

Executar Passo 11.

Passo 9 – Exibe opções do Menu de Consultas.

Executar Passo 20.

Passo 10 – Exibe opções do Menu de Relatórios.

Executar Passo 27.

Passo 11 – Opção escolhida é Arquivos de Fiscalização?

Não, executar Passo 12.

Sim, extrair os arquivos de Fiscalização.

Retorna Passo 4.

Passo 12 – Opção escolhida é Tipos de Usuários?

Não, executar Passo 13.

Sim, exibir a interface gráfica para cadastro e alteração de tipos de usuários.

Retorna Passo 4.

Passo 13 – Opção escolhida é Grupos de Usuários?

Não, executar Passo 14.

Sim, exibir a interface gráfica para cadastro e alteração de grupos de usuários.

Retorna Passo 4.

Passo 14 – Opção escolhida é Tipos de Identificação?

Não, executar Passo 15.

Sim, exibir a interface gráfica para cadastro e alterações de tipos de identificação.

Retorna Passo 4.

Passo 15 – Opção escolhida é Usuários?

Não, executar Passo 16.

Sim, exibir a interface gráfica para cadastro e alterações de usuários.

Retorna Passo 4.

Passo 16 – Opção escolhida é Mensagens Gerais?

Não, executar Passo 17.

Sim, exibir a interface gráfica para cadastro de mensagens gerais.

Retorna Passo 4.

Passo 17 – Opção escolhida é Tipos de Mensagens?

Não, executar Passo 18.

Sim, exibir a interface gráfica para cadastro e alterações de tipos de mensagens.

Retorna Passo 4.

Passo 18 – Opção escolhida é Tipos de Identif. Registros?

Não, executar Passo 19.

Sim, exibir a interface gráfica para cadastro e alterações de tipos de identificação de registros.

Retorna Passo 4.

Passo 19 – Opção escolhida é Tipos de Operações?

Não, executar Passo 20.

Sim, exibir a interface gráfica para cadastro e alterações de tipos de operações.

Retorna Passo 4.

Passo 20 – Opção escolhida é Tipos de Usuários?

Não, executar Passo 21.

Sim, exibir a interface gráfica para consulta e exclusão de tipos de usuários.

Retorna Passo 4.

Passo 21 – Opção escolhida é Grupos de Usuários?

Não, executar Passo 22.

Sim, exibir a interface gráfica para consulta e exclusão de grupos de usuários.

Retorna Passo 4.

Passo 22 – Opção escolhida é Tipos de Identificações?

Não, executar Passo 23.

Sim, exibir a interface gráfica para consulta e exclusão de tipos de identificações.

Retorna Passo 4.

Passo 23 – Opção escolhida é Usuários?

Não, executar Passo 24.

Sim, exibir a interface gráfica para consulta e exclusão de usuários.

Retorna Passo 4.

Passo 24 – Opção escolhida é Tipos de Identif. de Registros?

Não, executar Passo 25.

Sim, exibir a interface gráfica para consulta e exclusão de tipos de identificação de registros. Retorna Passo 4.

Passo 25 – Opção escolhida é Tipos de Operações?

Não, executar Passo 26.

Sim, exibir a interface gráfica para consulta e exclusão de tipos de operações.

Retorna Passo 4.

Passo 26 – Opção escolhida é Tipos de Mensagens?

Não, executar Passo 28.

Sim, exibir a interface gráfica para consulta de tipos de mensagens.

Retorna Passo 4.

Passo 27 – Opção escolhida é Espelho de Ponto?

Não, executar Passo 28.

Sim, extrair o relatório de Espelho de Ponto.

Retorna Passo 4.

Passo 28 – Efetuar o *logout*.

Finaliza a operação.

O fluxograma analítico que representa essas operações é representado na Figura 8. A operação em modo de acesso remoto, tem por objetivo permitir a manutenção dos dados cadastrais, bem como, a geração dos relatórios de espelho de ponto. A operação é efetuada conforme o modo de acesso local, porém, uma vez autenticado, o programa de gerenciamento remoto habilita os menus de cadastros, consultas e relatórios.

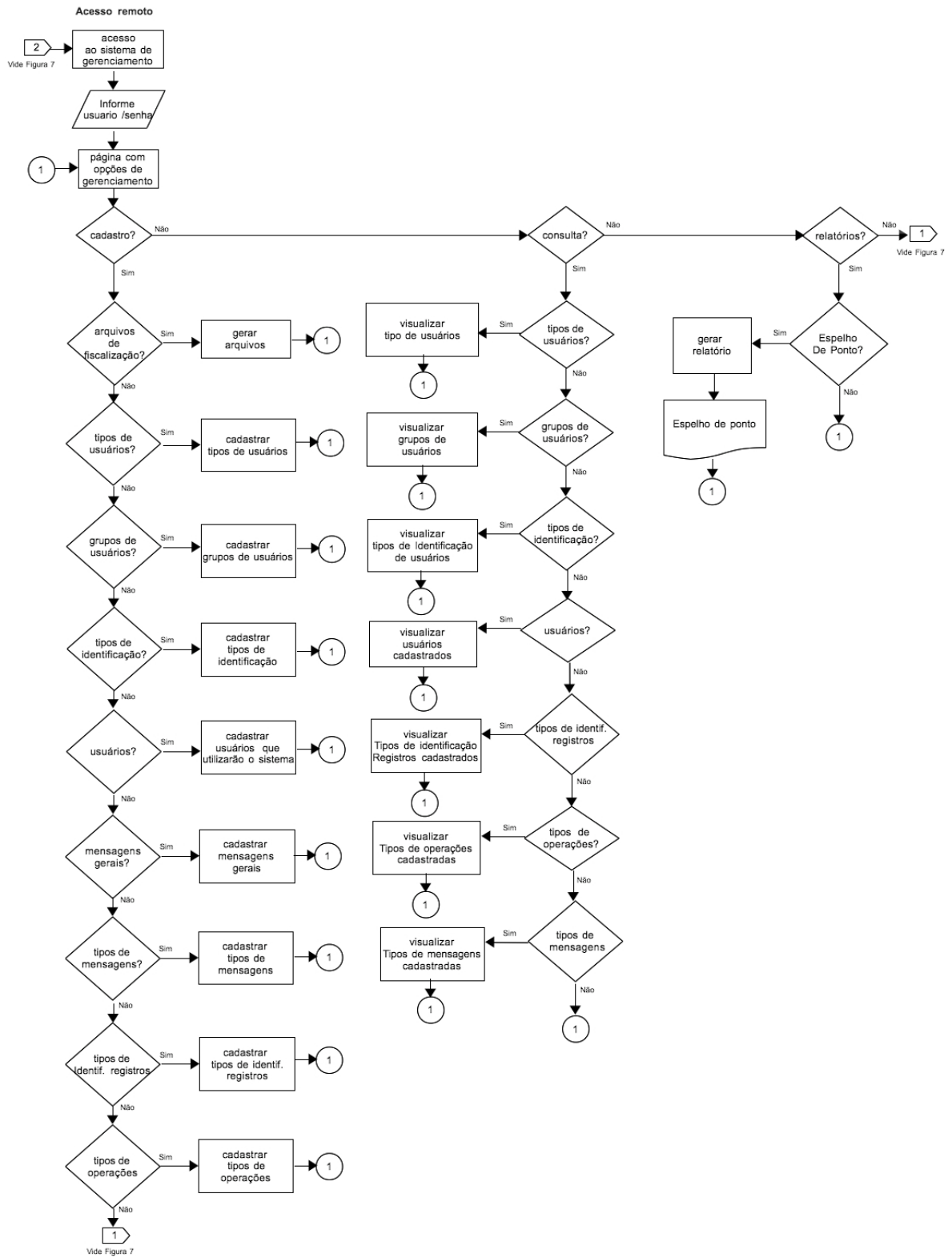


Figura 8: Fluxograma de acesso remoto

3.4.3 Interface gráfica

Nessa subsecção são apresentadas as interfaces gráficas obtidas durante diferentes etapas de execução do *software* em questão. As interfaces gráficas disponíveis no sistema proposto são auto-explicativas e conseqüentemente, intuitivas.

As interfaces correspondentes às funções de autenticação, cadastro, atualização, consulta, exclusão e relatórios, são apresentadas a seguir da seguinte maneira: registro do ponto, inicial, acesso, menu principal, cadastros, consultas e relatórios com respectivas descrições.

A Figura 9 apresenta a hierarquia de interfaces disponíveis no sistema e que estão representadas nessa subsecção.

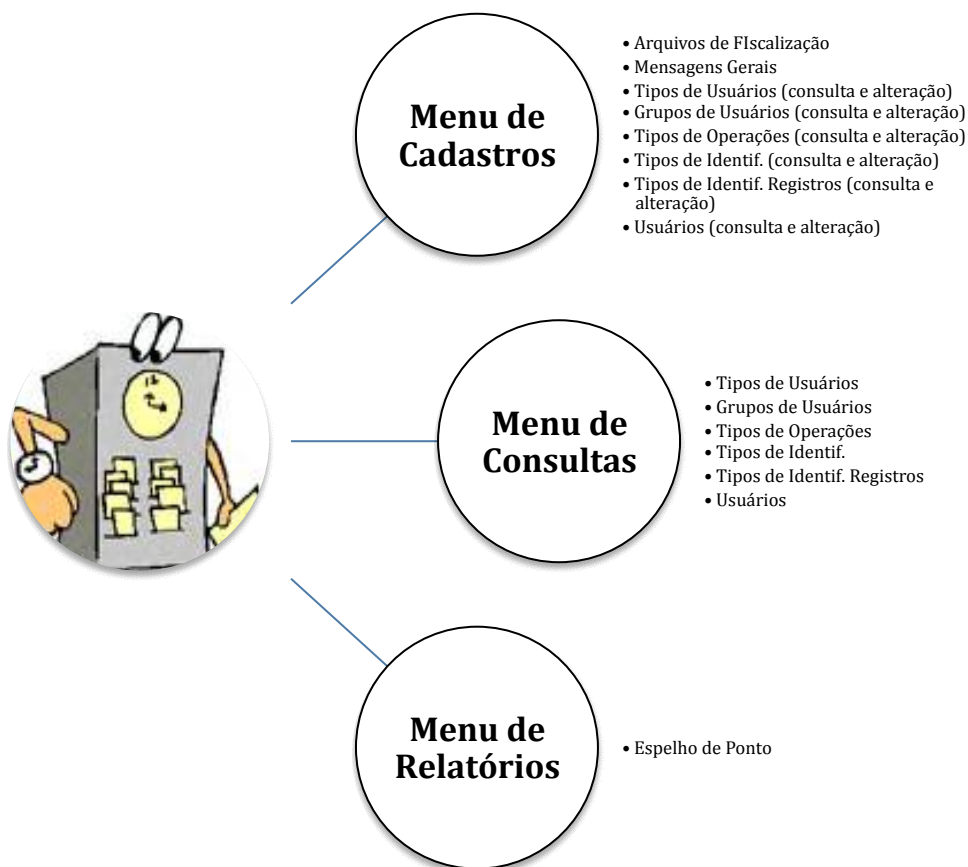


Figura 9: Hierarquia de Interfaces disponíveis no sistema – módulo remoto

A Figura 10 apresenta a hierarquia das interfaces disponíveis no módulo local para efetuar o registro eletrônico de ponto. As interfaces destacadas estão representadas nessa subseção.

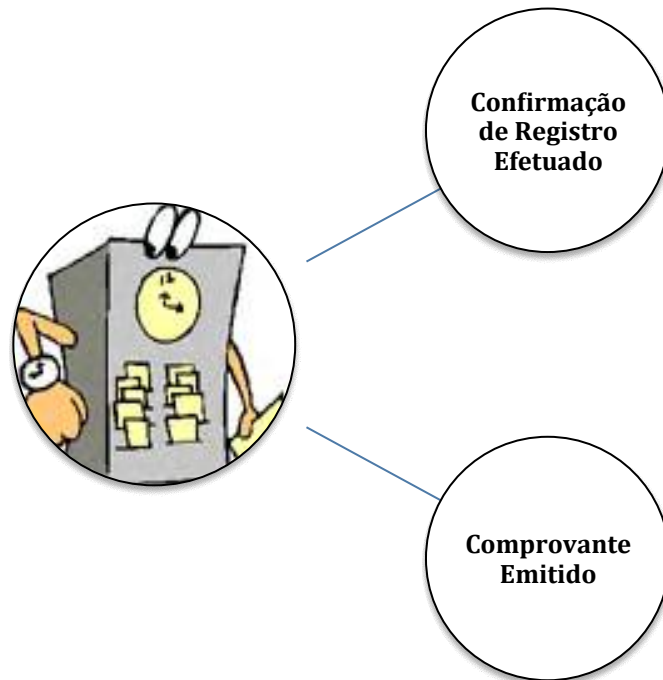


Figura 10: Hierarquia de Interfaces disponíveis no sistema – módulo local

A Figura 11 mostra a interface utilizada pelo usuário no módulo local de registro eletrônico de ponto, sendo necessária efetuar a autenticação do usuário. Após a validação do acesso do usuário, o comprovante de entrada e/ou saída é gerado automaticamente e direcionado para a impressora conectada ao sistema, conforme apresentado na arquitetura proposta.

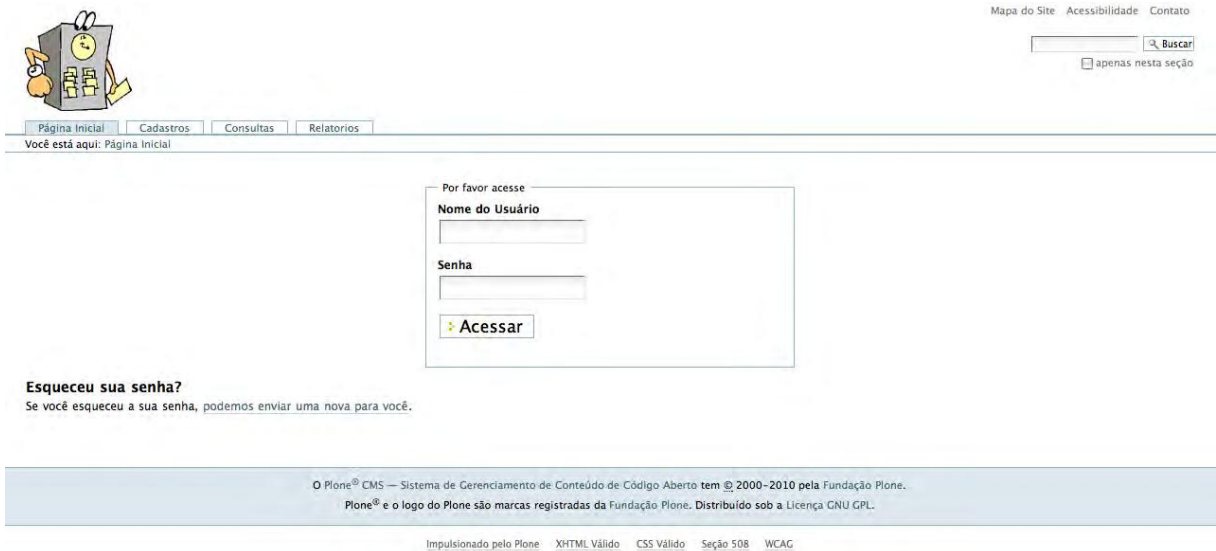


Figura 11: Interface gráfica do sistema de registro eletrônico de ponto

A Figura 12 apresenta a interface gráfica inicial disponível para qualquer usuário e para o administrador com possibilidade de acesso remoto. Nesse contexto é necessária uma autenticação para que o usuário tenha acesso a informações de registro de ponto, informações essas consideradas confidenciais e de visualização controlada.



Figura 12: Interface gráfica do sistema remoto – interface inicial

A Figura 13 mostra o usuário, já autenticado, tendo acesso aos módulos disponíveis de cadastro, consultas e relatórios no menu principal do sistema. O acesso é feito remotamente.

Na parte superior direita da Figura 13 é possível observar o nome do usuário autenticado e a opção para efetuar saída do sistema de gerenciamento remoto.

Mapa do Site Acessibilidade Contato

Buscar no Site apenas nesta seção

Página Inicial Cadastros Consultas Relatorios

Você está aqui: Página Inicial Ana Paula Muller Giancoli Sair

Sistema de Registro Eletronico com Gerenciamento Remoto

— última modificação 01/03/2010 00:59

No dia 21 de agosto de 2009, o Ministério do Trabalho e Emprego, ou simplesmente MTE, publicou a portaria número 1510 que tem como objetivo disciplinar o registro eletrônico de ponto e a utilização do Sistema de Registro Eletrônico de Ponto. Baseado nesta portaria, o sistema em questão foi desenvolvido para atendê-la utilizando software de código aberto e de custo acessível. Além de permitir o gerenciamento remoto das funcionalidades, bem como a geração dos relatórios de Espelho de Ponto.

Título	Autor	Tipo	Modificado
Cadastros	admin	Pasta	20/08/2010 15:29
Consultas	admin	Pasta	06/03/2010 15:35
Relatorios	admin	Pasta	06/03/2010 15:35

« Agosto 2010 »

Se	Te	Qu	Qu	Se	Sa	Do
						1
2	3	4	5	6	7	8
9	10	11	12	13	14	15
16	17	18	19	20	21	22
23	24	25	26	27	28	29
30	31					


Enviar — Imprimir —

O Plone® CMS — Sistema de Gerenciamento de Conteúdo de Código Aberto tem © 2000-2010 pela Fundação Plone.
Plone® e o logo do Plone são marcas registradas da Fundação Plone. Distribuído sob a Licença GNU GPL.

[Impulsionado pelo Plone](#) [XHTML Válido](#) [CSS Válido](#) [Seção 508](#) [WCAG](#)

Figura 13: Interface gráfica do sistema remoto – menu principal

A Figura 14 mostra um usuário já autenticado, com acesso aos menus de cadastros disponíveis no sistema. Por meio deste é possível fazer o acesso a opção de extração de dados referentes aos registros eletrônicos de ponto exigidos pela portaria 1510 e efetuar os cadastros relativos a usuários, grupos de usuários, mensagens gerais, tipos de identificação de registros e usuários, tipos de operações, usuários e mensagens.



Mapa do Site Acessibilidade Contato

Buscar no Site apenas nesta seção

Página Inicial Cadastros Consultas Relatorios

Você está aqui: Página Inicial → Cadastros

Ana Paula Muller Giancoli Sair

Você está aqui: Página Inicial → Cadastros

Navegação

- Arquivos de Fiscalizacao
- Grupos de Usuarios
- Mensagens Gerais
- Tipos de Identificacao
- Tipos de Identificacao de Registros
- Tipos de Mensagens
- Tipos de Operacoes
- Tipos de Usuarios
- Usuarios

Cadastros

por **admin** — última modificação 20/08/2010 15:29
Nenhum álbum ou fotografia foi enviado ainda.

Arquivos de Fiscalizacao — por **admin** — última modificação 28/08/2010 21:04
Gerar arquivos da Fiscalizacao

Grupos de Usuarios — por **admin** — última modificação 20/08/2010 01:43
Permite a manutenção de grupos de usuarios no Sistema.

Mensagens Gerais — por **admin** — última modificação 13/08/2010 14:36
Cadastro de Mensagens (email) gerais. Podendo ser de sistema ou entre usuários.

Tipos de Identificacao — por **admin** — última modificação 19/08/2010 22:18
Permite a manutenção de tipos de identificacao do usuario no Sistema.

Tipos de Identificacao de Registros — por **admin** — última modificação 19/08/2010 21:59
Permite a manutenção de tipos de identificacao dos registros dos arquivos da Fiscalizacao.

Tipos de Mensagens — por **admin** — última modificação 19/08/2010 21:37
Permite a manutenção de mensagens do Sistema.

Tipos de Operacoes — por **admin** — última modificação 19/08/2010 20:56
Permite a manutenção de tipos de operacao a serem identificados nos registros de fiscalizacao.

Tipos de Usuarios — por **admin** — última modificação 19/08/2010 21:24
Permite a manutenção de tipos de usuarios do Sistema.

Usuarios — por **admin** — última modificação 20/08/2010 15:03
Permite o cadastramento de Usuários do Sistema

« Agosto 2010 »

Se	Te	Qu	Qu	Se	Sa	Do
						1
2	3	4	5	6	7	8
9	10	11	12	13	14	15
16	17	18	19	20	21	22
23	24	25	26	27	28	29
30	31					

Enviar — Imprimir —

Figura 14: Interface gráfica do menu cadastros

A Figura 15 mostra o cadastro de tipos de usuários, no qual é possível inserir novos tipos de usuários, registrando o usuário e data de criação do registro no banco de dados do servidor (SERV.) representado na Figura 1.

Página Inicial Cadastros Consultas Relatorios

Você está aqui: Página Inicial → Cadastros → Tipos de Usuarios

Ana Paula Muller Giancoli Sair

Você está aqui: Página Inicial → Cadastros → Tipos de Usuarios

Navegação

- Arquivos de Fiscalizacao
- Grupos de Usuarios
- Mensagens Gerais
- Tipos de Identificacao
- Tipos de Identificacao de Registros
- Tipos de Mensagens
- Tipos de Operacoes
- Tipos de Usuarios**
- Usuarios

Tipos de Usuarios

Permite a manutenção de tipos de usuarios do Sistema.

Preencha os dados abaixo:

Dados de Identificacao

Dados de Identificacao do usuario que esta efetuando o cadastro

Identificacao Registro
Numero de Identificacao do registro a ser criado. Mantenha -1, caso deseje inserir um novo registro. Caso deseje atualizar algum registro, informe somente o identificador do registro.

Data de Criacao
Busca automaticamente a data de criacao do registro

Usuario de Criacao
Busca automaticamente o usuario de criacao do registro

Dados Tipos de Usuarios

Descricao Tipo de Usuarios
Entre com a descricao do tipo de usuarios: 1-Administrativo, 2-Empregador, 3-Professor, 4-Tecnico.

« Setembro 2010 »

Se	Te	Qu	Qu	Se	Sa	Do
		1	2	3	4	5
6	7	8	9	10	11	12
13	14	15	16	17	18	19
20	21	22	23	24	25	26
27	28	29	30			

Figura 15: Interface gráfica do cadastro de tipos de usuários

A Figura 16 mostra um usuário já autenticado, com acesso aos menus de atualização de tipos de usuários para editar as propriedades de um cadastro desse tipo, que já tenha sido armazenado no sistema.

The screenshot shows a web application interface for editing user types. The page title is "Tipos de Usuarios Update" and it includes a navigation menu, a breadcrumb trail, and a calendar widget. The main content area contains form fields for "Dados de Identificacao", "Identificacao Registro", "Data de Criacao", "Usuario de Criacao", and "Dados Tipos de Usuarios".

Navegação

- Grupo de Usuarios
- Tipos de Status
- Tipos de Usuarios
- Update

Você está aqui: Página Inicial → Atualizacoes → Tipos de Usuarios Update

Tipos de Usuarios Update
Permite a manutenção de tipos de usuarios do Sistema.

Preencha os dados abaixo:

Dados de Identificacao
Dados de Identificacao do usuario que esta efetuando o cadastro

Identificacao Registro
Numero de Identificacao do registro a ser criado. Mantenha -1, caso deseje inserir um novo registro. Caso deseje atualizar algum registro, informe o identificador do registro e preencha os demais campos a serem atualizados.

Data de Criacao
Busca automaticamente a data de criacao do registro

Usuario de Criacao
Busca automaticamente o usuario de criacao do registro

Dados Tipos de Usuarios

Descricao Tipo de Usuarios
Entre com a descricao do tipo de usuarios: 1-Administrativo, 2-Empregador, 3-Professor, 4-Tecnico.

Calendar: < Setembro 2010 >

Se	Te	Qu	Qu	Se	Sa	Do
		1	2	3	4	5
6	7	8	9	10	11	12
13	14	15	16	17	18	19
20	21	22	23	24	25	26
27	28	29	30			

Figura 16: Interface gráfica de edição de tipos de usuários

A Figura 17 mostra um usuário já autenticado, com acesso aos menus de consultas. Por meio desse menu é possível fazer o acesso as consultas relativas a tipos de usuários, grupos de usuários, tipos de operação, tipos de identificação, tipos de identificação de registros, tipos de mensagens e usuários.

Mapa do Site Acessibilidade Contato

Buscar no Site apenas nesta seção

Página Inicial Cadastros Consultas Relatorios

Você está aqui: Página Inicial → Consultas

Ana Paula Muller Giancoli Sair

Navegação

- Tipos de Usuarios
- Grupos de Usuarios
- Tipos de Operacoes
- Tipos de Identificacao
- Tipos de Identificacao de Registros
- Tipos de Mensagens
- Usuarios

Consultas

por [admin](#) — última modificação 06/03/2010 15:35
Nenhum álbum ou fotografia foi enviado ainda.

[Tipos de Usuarios](#) — por [admin](#) — última modificação 20/08/2010 16:04

[Grupos de Usuarios](#) — por [admin](#) — última modificação 20/08/2010 16:01

[Tipos de Operacoes](#) — por [admin](#) — última modificação 20/08/2010 16:30

[Tipos de Identificacao](#) — por [admin](#) — última modificação 20/08/2010 16:22

[Tipos de Identificacao de Registros](#) — por [admin](#) — última modificação 20/08/2010 16:41

[Tipos de Mensagens](#) — por [admin](#) — última modificação 20/08/2010 16:48

[Usuarios](#) — por [admin](#) — última modificação 20/08/2010 17:04

Enviar — Imprimir —

« Agosto 2010 »

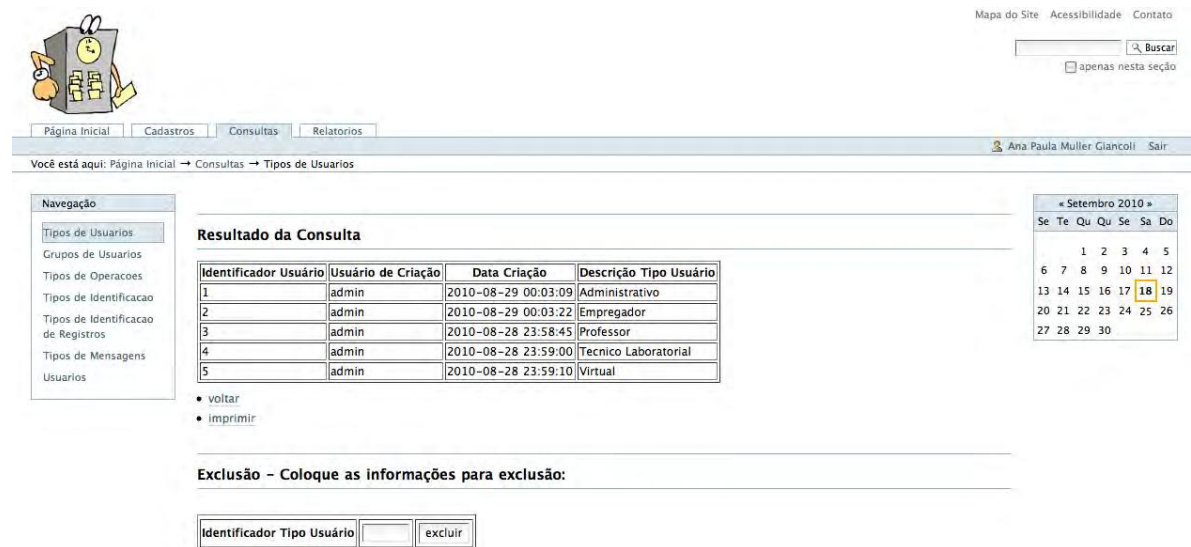
Se	Te	Qu	Qu	Se	Sa	Do
						1
2	3	4	5	6	7	8
9	10	11	12	13	14	15
16	17	18	19	20	21	22
23	24	25	26	27	28	29
30	31					

O Plone® CMS — Sistema de Gerenciamento de Conteúdo de Código Aberto tem © 2000-2010 pela Fundação Plone.
Plone® e o logo do Plone são marcas registradas da Fundação Plone. Distribuído sob a Licença GNU GPL.

Impulsionado pelo Plone XHTML Válido CSS Válido Seção 508 WCAG

Figura 17. Interface gráfica do menu de consultas

A Figura 18 mostra um usuário já autenticado, com acesso aos menus de consulta de tipos de usuários. É apresentada a lista de tipos de usuários cadastrados no sistema com as suas devidas descrições. Por meio dessa janela é possível excluir um tipo de usuário, desde que o mesmo não tenha sido utilizado anteriormente. Essa integridade é mantida pelo SGBD.



Mapa do Site Acessibilidade Contato

Buscar

apenas nesta seção

Página Inicial Cadastros Consultas Relatorios

Ana Paula Muller Giancoli Sair

Você está aqui: Página Inicial → Consultas → Tipos de Usuários

Navegação

- Tipos de Usuários
- Grupos de Usuarios
- Tipos de Operacoes
- Tipos de Identificacao
- Tipos de Identificacao de Registros
- Tipos de Mensagens
- Usuarios

Resultado da Consulta

Identificador Usuário	Usuário de Criação	Data Criação	Descrição Tipo Usuário
1	admin	2010-08-29 00:03:09	Administrativo
2	admin	2010-08-29 00:03:22	Empregador
3	admin	2010-08-28 23:58:45	Professor
4	admin	2010-08-28 23:59:00	Tecnico Laboratorial
5	admin	2010-08-28 23:59:10	Virtual

- voltar
- imprimir

Exclusão - Coloque as informações para exclusão:

Identificador Tipo Usuário

« Setembro 2010 »

Se	Te	Qu	Qu	Se	Sa	Do
		1	2	3	4	5
6	7	8	9	10	11	12
13	14	15	16	17	18	19
20	21	22	23	24	25	26
27	28	29	30			

Figura 18. Interface gráfica utilizada para consultar os tipos de usuários

A Figura 19 mostra um usuário já autenticado, com acesso aos menus de relatórios. O usuário poderá extrair o relatório de espelho de ponto disponível nessa janela.



Mapa do Site Acessibilidade Contato

Buscar no Site Buscar

apenas nesta seção

Página Inicial Cadastros Consultas Relatorios

Ana Paula Muller Giancoli Sair

Você está aqui: Página Inicial → Relatorios

Navegação

- Espelho de Ponto

Relatorios

por [admin](#) — última modificação 06/03/2010 15:35
Nenhum álbum ou fotografia foi enviado ainda.

[Espelho de Ponto](#) — por [admin](#) — última modificação 31/08/2010 20:27

Enviar — Imprimir —

« Agosto 2010 »

Se	Te	Qu	Qu	Se	Sa	Do
						1
2	3	4	5	6	7	8
9	10	11	12	13	14	15
16	17	18	19	20	21	22
23	24	25	26	27	28	29
30	31					

O Plone® CMS — Sistema de Gerenciamento de Conteúdo de Código Aberto tem © 2000-2010 pela Fundação Plone.
Plone® e o logo do Plone são marcas registradas da Fundação Plone. Distribuído sob a Licença GNU GPL.

Impulsionado pelo Plone XHTML Válido CSS Válido Seção 508 WCAG

Figura 19. Interface gráfica do menu de relatórios

3.4.4 Testes funcionais

Na realização dos testes práticos, necessários para validar a arquitetura proposta neste trabalho, foi implementada uma arquitetura de *hardware* e *software* apresentada na Figura 6, a qual é capaz de atender as ações contidas no fluxograma analítico mostrado na Figura 7 e Figura 8.

A execução dos testes foi dividida em duas etapas, sendo que na primeira foi avaliada a funcionalidade de registro eletrônico de ponto com uma arquitetura de acesso local e operação dedicada. Na segunda etapa de testes, foram realizados os cadastros de usuários e grupos, necessários para extração dos relatórios de espelho de ponto, funcionalidade essa disponível no módulo de gerenciamento e acesso remoto.

3.4.4.1 Acesso local e operação dedicada

Para ter acesso ao registro eletrônico de ponto, é necessário que o usuário do sistema informe a identificação e senha, conforme apresentado na Figura 11.

O recurso de identificação de usuário e senha estabelece o controle de segurança mínimo para que somente os usuários previamente cadastrados na memória de trabalho (MT) do sistema possam efetuar o registro de ponto, conforme a Figura 20.

numidentif	nropispasep	nrocei	iduser	loguser	nomecompleto	localserv	nrofabricacao
959878786000	959878786000	876666544444	12	relponto	IFSP Relogio Ponto Eletronico	Braganca Paulista	FG34dew345
00462469670	126201289868		14	agiancoli	Ana Paula Muller Giancoli	Braganca Paulista	
34561234567	126201289868		18	mmacedo	Mariana Macedo	Braganca Paulista	
12376543218	888888888812		19	abrega	Ana Maria da Silva Brega	Braganca Paulista	

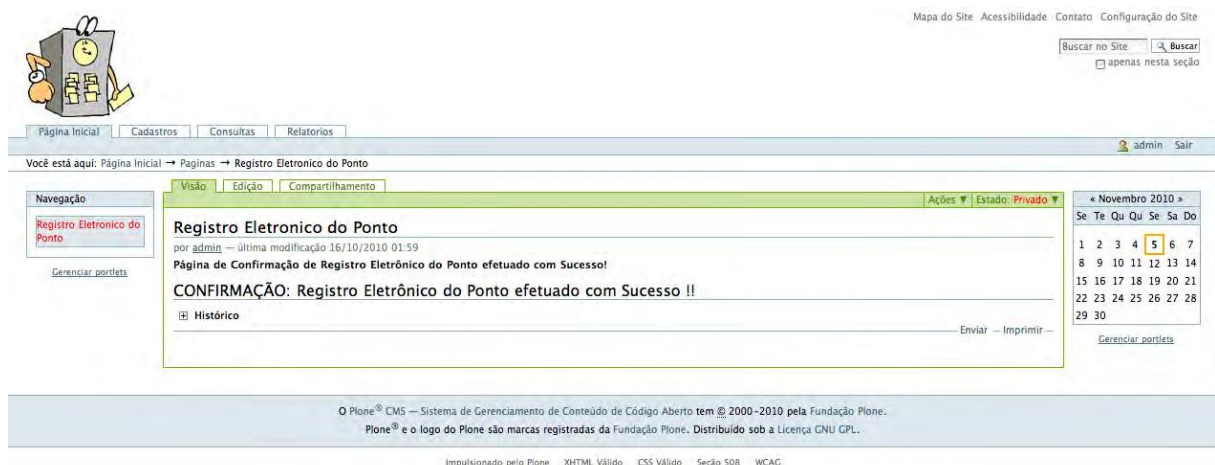
Figura 20: Informações de usuários registradas na memória de trabalho

Após o usuário informar os dados de usuário e senha válidas, as informações necessárias para identificação do registro eletrônico de ponto são gravadas na memória permanente (MRP), conforme mostrado na Figura 21. Essas informações são utilizadas na geração do relatório de espelho do ponto, pois é possível recuperar o nome do usuário e os horários registrados de trabalho.

ididentreg	idtpoper	idtpidentif	numidentif	nomecompleto	nropispasep	localserv	nrocei	dataregistro
2	1	2	462469670	Ana Paula Muller Giancoli	126201289868	Braganca Paulista		2010-09-18 10:36:16
3	2	1	NULL	NULL	NULL	NULL	NULL	2010-09-18 10:38:07
2	1	2	12376543218	Ana Maria da Silva Brega	888888888812	Braganca Paulista		2010-09-18 10:39:30
3	2	1	NULL	NULL	NULL	NULL	NULL	2010-09-18 12:16:21
2	1	2	462469670	Ana Paula Muller Giancoli	126201289868	Braganca Paulista		2010-09-18 12:19:30
2	1	2	12376543218	Ana Maria da Silva Brega	888888888812	Braganca Paulista	NULL	2010-09-18 20:30:03

Figura 21: Informações registradas na memória permanente

Em seguida, o sistema apresenta a janela principal com a mensagem de registro eletrônico efetuado com sucesso, conforme mostrado na Figura 22.



Mapa do Site Acessibilidade Contato Configuração do Site

Buscar no Site Buscar apenas nesta seção

Página Inicial Cadastros Consultas Relatórios

Você está aqui: Página Inicial → Páginas → Registro Eletrônico do Ponto

admin Sair

Navegação

Registro Eletrônico do Ponto

Gerenciar portlets

Visão Edição Compartilhamento

Ações Estado: Privado

« Novembro 2010 »

Se	Te	Qu	Qu	Se	Sa	Do
1	2	3	4	5	6	7
8	9	10	11	12	13	14
15	16	17	18	19	20	21
22	23	24	25	26	27	28
29	30					

Gerenciar portlets

O Plone® CMS — Sistema de Gerenciamento de Conteúdo de Código Aberto tem © 2000-2010 pela Fundação Plone.
Plone® e o logo do Plone são marcas registradas da Fundação Plone. Distribuído sob a Licença GNU GPL.

Impulsionado pelo Plone XHTML Válido CSS Válido Seção 508 WCAG

Figura 22: Interface gráfica de confirmação de registro eletrônico de ponto

Na Figura 23 é apresentado o comprovante de registro eletrônico de ponto emitido por meio de impressora acoplada ao equipamento local com informações fictícias.

Retorna à interface gráfica de acesso, conforme apresentada na Figura 11 para que o próximo usuário efetue o registro eletrônico do ponto.

COMPROVANTE DE REGISTRO DE PONTO DO TRABALHADOR	
Empregador	Nome: Empresa XXX CPF/CNPJ: 08.585.987/0001-77 CEI: (não consta) Local Prestação Serviço: São Paulo Nro.Fab.: 87HT543CX
Empregado	Nome: Ana Paula Müller Giancoli PIS: 123.4563.98-90 NSR: 2
Registro em 16/10/10 14:44	

Figura 23: Comprovante de registro eletrônico de ponto

3.4.4.2 Acesso remoto e módulo de gerenciamento

Para ter acesso ao módulo de gerenciamento e acesso remoto, é necessário que o usuário do sistema informe a identificação e senha, conforme apresentado na Figura 24.

Mapa do Site | Acessibilidade | Contato

Buscar no Site apenas nesta seção

Página Inicial | Cadastros | Consultas | Relatórios

Você está aqui: Página Inicial

Acessar

Nome do Usuário

Senha

Sistema de Registro Eletronico com Gerenciamento Remoto

No dia 21 de agosto de 2009, o Ministério do Trabalho e Emprego, ou simplesmente MTE, publicou a portaria número 1510 que tem como objetivo disciplinar o registro eletrônico de ponto e a utilização do Sistema de Registro Eletrônico de Ponto. Baseado nesta portaria, o sistema em questão foi desenvolvido para atendê-la utilizando software de código aberto e de custo acessível. Além de permitir o gerenciamento remoto das funcionalidades, bem como a geração dos relatórios de Espelho de Ponto.

Título	Tipo
Cadastros	Pasta
Consultas	Pasta
Relatórios	Pasta

« Agosto 2010 »

Se	Te	Qu	Se	Sa	Do
					1
2	3	4	5	6	7
8	9	10	11	12	13
14	15	16	17	18	19
20	21	22	23	24	25
26	27	28	29	30	31

O Plone® CMS — Sistema de Gerenciamento de Conteúdo de Código Aberto tem © 2000–2010 pela Fundação Plone.
Plone® e o logo do Plone são marcas registradas da Fundação Plone. Distribuído sob a Licença GNU GPL.

Impulsionado pelo Plone | XHTML Válido | CSS Válido | Seção 508 | WCAG

Figura 24: Interface gráfica do sistema de registro - acesso remoto

Esse recurso estabelece o controle de segurança mínimo para que somente os usuários previamente cadastrados na memória de trabalho (MT) do sistema, conforme apresentado na Figura 20, possam efetuar o registro do ponto por meio da interface gráfica.

Após o usuário informar os dados de usuário e senha válidas, o sistema apresenta a janela com as opções de acesso a menus conforme Figura 12.

Permite que os usuários cadastrados no sistema possam extrair o relatório de espelho de ponto por meio da opção de menu relatórios.

Esse recurso apresentado na Figura 14, permite ao usuário administrador efetuar diversos cadastros, principalmente, de novos usuários para que os mesmos possam efetuar o registro eletrônico de ponto.

Na opção de cadastro de:

Tipos de usuários, são inseridas os tipos de usuários pré-definidos que poderão acessar o sistema, como por exemplo, Empregador, Empregado, Administrativo etc.

Grupos de usuários, são inseridos os grupos de usuários que utilizarão o sistema, como por exemplo, Técnicos, Administradores etc.

Tipos de identificação, são inseridos os tipos de documentos que poderão ser utilizados como identificadores do usuário, como por exemplo, CNPJ, CPF, PIS, *login* etc.

Usuários, são cadastrados os usuários que poderão efetuar o registro eletrônico de ponto, bem como o acesso remoto para extração do relatório de espelho de ponto, conforme previsto na portaria 1510.

Mensagens, são cadastradas as mensagens de emails para serem enviadas internamente aos usuários do sistema.

Tipos de identificação de registros, são inseridos os tipos de registros que são modificados. Ao incluir ou alterar informações de registros do tipo empregador, registros do tipo empregado, registros do tipo de marcação de ponto e registros de ajuste de relógio, os dados devem ser atualizados na MRP, identificados conforme o tipo de registro alterado.

Tipos de operação, são inseridas as operações possíveis, como por exemplo, inclusão, atualização ou exclusão de informações pertinentes ao empregador, empregado, ajuste do relógio e marcação de ponto.

Na Figura 25 é apresentado a interface gráfica de geração de arquivos de fiscalização,

referentes às marcações dos horários de entrada e saída dos funcionários, exigidos pela portaria.

The screenshot shows a web application interface for generating fiscalization files. The interface is titled "Arquivos de Fiscalizacao" and contains several sections for data entry and selection.

Arquivos de Fiscalizacao

Gerar arquivos da Fiscalizacao

Dados de Identificacao

Dados de Identificacao do usuario que esta efetuando o cadastro

Data de Criacao

Busca automaticamente a data de criacao do registro

2010 / outubro / 16 : 15 : 20

Usuario de Criacao

Busca automaticamente o usuario de criacao do registro

admin

Arquivos de Fiscalizacao

Selecione os arquivos de fiscalizacao que deseja gerar

IdEmpregador

Informe o Id do Empregador

1

AFD: ARQUIVO-FONTE DE DADOS

AFDT: ARQUIVO-FONTE DE DADOS TRATADO

ACJEF: ARQUIVO DE CONTROLE DE JORNADA PARA EFEITOS FISCAIS

Periodo de Geracao

Selecione o periodo para geracao dos arquivos

Periodo Inicial

Informe o periodo inicial para geracao dos arquivos

2010 / outubro / 16

Periodo Final

Informe o periodo final para geracao dos arquivos

2010 / outubro / 16

Confirmar Limpar

Calendar: Se Te Qu Qu Se Sa Do

					1	2	3
4	5	6	7	8	9	10	
11	12	13	14	15	16	17	
18	19	20	21	22	23	24	
25	26	27	28	29	30	31	

Gerenciar portlets

Figura 25: Interface gráfica para geração dos arquivos da fiscalização

3.4.5 Resultados

Os resultados obtidos com o protótipo do *software* foram satisfatórios, pois atenderam aos requisitos de *software* listados e identificados como aptos nas Tabelas 1 e 2 da subseção 3.2. Os fatores levados em consideração nessa avaliação foram os quesitos de desenho e usabilidade da interface gráfica, a funcionalidade oferecida pela solução e sua correspondência com os requisitos exigidos pelo portaria numero 1510 e as possibilidades de atualização e manutenção dos módulos que compõem a aplicação.

A proposta de arquitetura apresentada é viável para a aplicação a qual se destina e atende a maioria dos requisitos funcionais exigidos pela portaria que regulamenta o registro eletrônico de ponto, requisitos esses destacados nas Tabelas 1 e 2, da subseção 3.2. Além de ser baseada em *software* livre a solução proposta também faz uso de tecnologias e linguagens com documentação disponível.

CAPÍTULO 4. CONCLUSÕES E TRABALHOS FUTUROS

Os sistemas baseados em *software* livre para registros eletrônicos de ponto disponíveis estão em sua grande maioria descontinuados e são anteriores à publicação da portaria do Ministério do Trabalho e Emprego de número 1510. As opções que supostamente continuam em desenvolvimento também não são compatíveis com essa portaria.

A arquitetura proposta nesse trabalho, no âmbito de *software* e *hardware*, é capaz de efetuar o registro eletrônico de ponto em conformidade com o previsto na mencionada portaria. Os recursos necessários para a utilização do sistema são triviais, não exigem interfaces gráficas e *softwares* de desenvolvimentos com esforço computacional significativo, ou nem mesmo um grande número de bibliotecas específicas. A interface com o usuário pode ser implementada mediante o uso de navegadores disponíveis no mercado, baseado em texto, por exemplo.

A validação do *software* referente à solução de *hardware* adotada como registrador eletrônico de ponto foi realizada em um computador local com o sistema operacional principal que reproduz os requisitos de *hardware* exigidos pela portaria de número 1510, situação que não impõe perda expressiva ao desempenho do sistema.

A solução proposta atende aos requisitos da portaria 1510, especificados nas Tabela 1 e 2, das subseção 3.2 e possibilita aos seus utilizadores independência de fornecedores de *software*, uma vez que o trabalho foi todo implementado utilizando *software* livre. Desse modo, futuras atualizações da portaria, com novas exigências, podem ser implementadas pela equipe de tecnologia da informação da empresa que adota essa solução ou ainda, por desenvolvedores terceirizados contratados sem a necessidade da dependência de um desenvolvedor específico responsável pela venda do equipamento e/ou sistema.

A maior dificuldade encontrada no desenvolvimento do trabalho foi a utilização ainda limitada da plataforma *Zope/Plone* nesse tipo de aplicação. Desse modo, a solução de problemas por meio de pesquisas ou de maneira autodidata foi de grande valia para o aprendizado das tecnologias envolvidas na confecção do trabalho. A obtenção de informações e trocas de experiências não foi realizada somente via consulta de livros ou material na internet, mas também com o contato direto com a comunidade de desenvolvedores de *python*

do Brasil no evento ocorrido no ano de 2010, sendo a décima primeira edição do Fórum Internacional de *Software* Livre, principal evento relacionado a essa temática no país.

A seqüência natural do presente trabalho é a implementação de uma solução de *hardware* integrada com o registrador eletrônico de ponto (REP). A solução apresentada por MONTEIRO (2008) poderá ser um ponto de partida, com adequações necessárias para o funcionamento. Reformulações na interface gráfica e adequações de funcionalidades são conseqüências e evoluções naturais, que deverão ocorrer após o retorno da comunidade de usuários, fato esse que possivelmente acontecerá assim que a aplicação se tornar amplamente utilizado.

REFERÊNCIAS

AGUIAR, T. L., S1 ponto, Outubro 2008. Disponível em <<http://sites.google.com/a/opens1.com/s1ponto/install>>. Consultado em 28 de setembro de 2010.

BARROS, J. R. M., Abril-Junho 1997. Avaliação do Processo de reestruturação Industrial Brasileira. Disponível em <http://geein.fclar.unesp.br/bibliografia/avaliacao_do_processo_de_reestruturacao_industrial.pdf>. Consultado em 11 de dezembro de 2010.

BOOCH, G.; RUMBAUGH, J.; JACOBSON I., Uml Guia do Usuário. Editora Campus, Segunda Edição, 2005.

BRASIL, Transporta - Portaria 1510, com novas normas para os relógios de ponto, deve ter grande impacto na economia brasileira, Fevereiro 2010. Disponível em <<http://www.transportabrasil.com.br/2010/02/portaria-1510-com-novas-normas-para-os-relogios-de-ponto-deve-ter-grande-impacto-na-economia-brasileira/>>. Consultado em 10 de dezembro de 2010.

DEBIAN. The universal operating system. Disponível em <<http://www.debian.org>>. Consultado em 28 de setembro de 2010.

FUZZER, I. B., wponto, Fevereiro 2009b. Disponível em <<http://www.ubuntero.com.br/?p=650>>. Consultado em 28 de setembro de 2010.

FUZZER, I. B., wponto, sistema de ponto eletrônico em php com suporte a vários bancos de dados, Junho 2009a. Disponível em <<http://sourceforge.net/projects/wponto/files/>>. Consultado em 28 de setembro de 2010.

G1, Adiado para março de 2011 o prazo para adoção do novo ponto eletrônico, Agosto 2010. Disponível em <<http://g1.globo.com/concursoseemprego/noticia/2010/08/mteadiaparamarcode2011oprazoparaadocaodonovopontoeletronico.html>>. Consultado em 28 de agosto de 2010.

LEAL, B. C., Sis ponto - sistema de ponto eletrônico via browser em php e postresql, Junho 2008. Disponível em <<http://sourceforge.net/projects/sisponto/>>. Consultado em 28 de setembro de 2010.

MONTEIRO, J., Fim do ponto eletrônico somente com software livre, Junho 2008. Disponível em <<http://www.guiadohardware.net/comunidade/ponto-montar/874769/>>. Consultado em 28 de setembro de 2010.

MTE, Ministério do Trabalho e Emprego, 21 de agosto de 2009, Portaria número 1510. Disponível em <http://www.mte.gov.br/legislacao/portarias/2009/p_20090821_1510.pdf>. Consultado em 28 de setembro de 2010.

MTE, Rep registrados no mte, Agosto 2010. Disponível em <http://www.mte.gov.br/ponto_eletronico/rep.asp>. Consultado em 28 de setembro de 2010.

MYSQL, MySQL 5.5. Disponível em <<http://www.mysql.org>>. Consultado em 28 de setembro de 2010.

PAPIZAN, K., Php timeclock, 01 2006. Disponível em <<http://timeclock.sourceforge.net>>. Consultado em 28 de setembro de 2010.

PLONE CMS, Open Source Content Manager. Disponível em <<http://www.plone.org>>. Consultado em 28 de setembro de 2010.

PYTHON PROGRAMMING LANGUAGE, Official Website. Disponível em <<http://www.python.org>>. Consultado em 28 de setembro de 2010.

STALLMAN, R. M., Free software foundation, Agosto 2010. Disponível em <<http://www.fsf.org>>. Consultado em 28 de setembro de 2010.

WAGNER, R., Fingerfx2 - sistema de gerenciamento de ponto eletrônico com suporte a biometria, Junho 2010. Disponível em <<http://www.linuxfx.org>>. Consultado em 28 de setembro de 2010.

ZOPE COMMUNITY, The Web Site for the Zope Community. Disponível em <<http://www.zope.org>>. Consultado em 28 de setembro de 2010.

APÊNDICE A – Codificação

SCRIPT DE REGISTRO ELETRÔNICO DO PONTO NO MÓDULO LOCAL

```
#!/usr/bin/python
# coding: utf-8

# Importando as bibliotecas necessarias
import os

#####
# Ajuste de Relogio de Ponto Eletronico #
#####
#           Ana Paula Muller Giancoli           #
#           Mestrado Unitau                       #
#           2011                                  #
#####

def abrirConexao (ip, usuario, senha, nomeDoBanco) :

    print
    print "Argumentos referentes a conexao: "
    print "----- "
    print "ip: " + ip
    print "usuario: " + usuario
    print "senha: " + "(omitida por seguranca)"
    print "nome do banco: " + nomeDoBanco
    print

    try :
        # Importando as bibliotecas necessarias
        import MySQLdb

        # Abrindo a conexao com o banco de dados
        conexao = MySQLdb.connect(ip, usuario, senha, nomeDoBanco)

        print
        print "*****"
        print "* Base de dados aberta com exito! *"
        print "*****"
        print

        # Retorna o objeto de conexao
        return conexao

    except :
        print
        print "*****"
        print "* Erro ao abrir a base de dados! *"
        print "*****"
        print

def fecharConexao (conexao) :
```

```

try :
    # Fechando a conexao com o banco
    conexao.commit()
    conexao.close()

    print
    print "*****"
    print "* Base de dados fechada com sucesso! *"
    print "*****"
    print

except :
    print
    print "*****"
    print "* Erro ao fechar a base de dados! *"
    print "*****"
    print

def consultarUsuarioMT (conexao, usuario) :

    try :
        # Definindo o cursor
        cursor = conexao.cursor()

        # Consultando o usuario e a senha
        sql = "SELECT loguser from mt"
        cursor.execute(sql)
        resultSet = cursor.fetchall()
        chaveUsuario = False

        for resultados in resultSet :
            if (resultados[0] == usuario) :
                chaveUsuario = True

        if (chaveUsuario == True) :
            print "Usuario logado com sucesso!"
        else :
            print "Dados invalidos!"

        return chaveUsuario

    except :
        print
        print "*****"
        print "* Erro ao consultar a base de dados! *"
        print "*****"
        print

def gravarRegistroMRP (conexao, usuario) :

    try :
        # Definindo o cursor
        cursor = conexao.cursor()

        # Trazendo os dados do usuario da MT
        sql = "SELECT * from mt WHERE loguser='" + usuario + "'"
        cursor.execute(sql)
        resultSet = cursor.fetchall()

        # Passando os dados para as variaveis
        for resultados in resultSet :
            idmt = resultados[0]

```

```

        ididentreg = resultados[1]
        idtpidentif = resultados[2]
        numidentif = resultados[3]
        nropispasep = resultados[4]
        nrocei = resultados[5]
        loguser = resultados[7]
        nomecompleto = resultados[8]
        localserv = resultados[9]
        nrofabricacao = resultados[10]

# print "idmt: " + str(idmt)
# print "ididentreg: " + str(ididentreg)
# print "idtpidentif: " + str(idtpidentif)
# print "numidentif: " + str(numidentif)
# print "nropispasep: " + str(nropispasep)
# print "nrocei: " + str(nrocei)
# print "loguser: " + str(loguser)
# print "nomecompleto: " + str(nomecompleto)
# print "localserv: " + str(localserv)
# print "nrpfabricacao: " + str(nrofabricacao)

# Gravando registro na MRP
try :
    sql = "INSERT INTO mrp (ididentreg, idtpoper, idtpidentif,
numidentif, nomecompleto, nropispasep, localserv, nrocei, idmt) VALUES (" +
str(ididentreg) + ", " + "1" + ", " + str(idtpidentif) + ", " + str(numidentif)
+ ", '" + str(nomecompleto) + "', " + str(nropispasep) + ", '" + str(localserv)
+ "', " + str(nrocei) + ", " + str(idmt) + ")"
        cursor.execute(sql)

    except :
        print "Dados nao gravados na MRP!"

    print
    print "*****"
    print "*   Dados salvos com sucesso na MRP   *"
    print "*****"
    print

except :
    print
    print "*****"
    print "*   Erro ao gravar na base de dados!   *"
    print "*****"
    print

# Inicio do programa principal

# Importando as bibliotecas necessarias
import os

os.system("clear")

usuario = raw_input("Digite seu nome: ")

conexao = abrirConexao("127.0.0.1","root","","RegistroPonto")
validador = consultarUsuarioMT(conexao, usuario)
if (validador == True) :
    gravarRegistroMRP(conexao, usuario)
else :
```

```

print
print "*****"
print "* Nenhum dado salvo na MRP *"
print "*****"
print
fecharConexao(conexao)

```

SCRIPT DE AJUSTE DE RELÓGIO NO MÓDULO LOCAL

```

#!/usr/bin/python
# coding: utf-8
# Importando as bibliotecas necessarias
import datetime
import os
import time

#####
# Ajuste de Relogio de Ponto Eletronico #
#####
# Ana Paula Muller Giancoli #
# Mestrado Unitau #
# 2011 #
#####

def abrirConexao (ip, usuario, senha, nomeDoBanco) :

    print
    print "Argumentos referentes a conexo: "
    print "----- "
    print "ip: " + ip
    print "usuario: " + usuario
    print "senha: " + "(omitida por seguranca)"
    print "nome do banco: " + nomeDoBanco
    print

    try :
        # Importando as bibliotecas necessarias
        import MySQLdb

        # Abrindo a conexao com o banco de dados
        conexao = MySQLdb.connect(ip, usuario, senha, nomeDoBanco)

        print
        print "*****"
        print "* Base de dados aberta com exito! *"
        print "*****"
        print

        # Retorna o objeto de conexao
        return conexao

    except :
        print
        print "*****"
        print "* Erro ao abrir a base de dados! *"

```

```

        print "*****"
        print

def fecharConexao (conexao) :

    try :
        # Fechando a conexao com o banco
        conexao.commit()
        conexao.close()

        print
        print "*****"
        print "* Base de dados fechada com sucesso! *"
        print "*****"
        print

    except :
        print
        print "*****"
        print "* Erro ao fechar a base de dados! *"
        print "*****"
        print

def consultarUsuarioMT (conexao, usuario) :

    try :
        # Definindo o cursor
        cursor = conexao.cursor()

        # Consultando o usuario e a senha
        sql = "SELECT loguser from mt"
        cursor.execute(sql)
        resultSet = cursor.fetchall()
        chaveUsuario = False

        for resultados in resultSet :
            if (resultados[0] == usuario) :
                chaveUsuario = True

        if (chaveUsuario == True) :
            print "Usuario logado com sucesso!"
        else :
            print "Dados invalidos!"

        return chaveUsuario

    except :
        print
        print "*****"
        print "* Erro ao consultar a base de dados! *"
        print "*****"
        print

def gravarRegistroMRP (conexao) :

    try :
        # Definindo o cursor
        cursor = conexao.cursor()

        from datetime import date
        hoje = date.today()

```

```

        agora = time.strftime('%H:%M:%S')

        print "Data armazenada no banco de dados: "
        print "----- "
        print hoje
        print "Hora armazenada no banco de dados: "
        print "----- "
        print agora

        sql = "INSERT INTO mrp (ididentreg, idtpoper, idtpidentif,
dataajuste, horaajuste, dataajustada, horaajustada, idmt) VALUES (3, 2, 1, " +
str(hoje) + ", '1', '1', '1', 5)"
        cursor.execute(sql)

        print
        print "*****"
        print "*   Dados salvos com sucesso na MRP   *"
        print "*****"
        print

    except :
        print
        print "*****"
        print "*   Erro ao gravar na base de dados!   *"
        print "*****"
        print

# Inicio do programa principal

try :
    os.system("clear")
    print "Verificando rede..."
    os.system("sudo /usr/sbin/ntpdate ntp.usp.br")
    os.system("/bin/date")
    print
    print "*****"
    print "*   Relogio atualizado com exito!   *"
    print "*****"
    print

except :
    print
    print "*****"
    print "*   Erro ao atualizar relógio!   *"
    print "*****"
    print

# Importando as bibliotecas necessarias
import os

os.system("clear")

conexao = abrirConexao("127.0.0.1", "root", "", "RegistroPonto")
gravarRegistroMRP(conexao)
fecharConexao(conexao)

```



```

from Products.PythonScripts.standard import url_quote
request = container.REQUEST
response = request.RESPONSE
form = request.form

try:
    if request.desctpuser==' ' and int(form.get('idtpuser', '-1')) < 0:
        destination = 'http://localhost:8080/Plone/paginas/erro'
        request.response.redirect(destination)
    else:
        if int(form.get('idtpuser', '-1')) >= 0:
            # if form.has_key('idtpuser'):
                res = context.testReadRow().dictionaries()
                if len(res) == 1:
                    row = res[0]
                    for key in row.keys():
                        form[key] = row[key]
                        id = row['idtpuser']
                        parameter = url_quote(id)
                        destination =
'http://localhost:8080/Plone/atualizacoes/Tipos de Usuarios Update'
                        request.response.redirect(destination + "?idtpuser=" +
parameter)
                    else:
context.testCreateRow(usercriacao=request.usercriacao,desctpuser=request.desctpu
ser)
                destination = 'http://localhost:8080/Plone/paginas/sucesso'
                request.response.redirect(destination)
except:
    destination = 'http://localhost:8080/Plone/paginas/erro'
    request.response.redirect(destination)

```

Script: testCreateRow
Parâmetros: usercriacao desctpuser
Função: Comando para inserção no banco de dados

```

insert into tipousuario(usercriacao,desctpuser) values (
    <dtml-sqlvar usercriacao type="string">,
    <dtml-sqlvar desctpuser type="string">
)

```

Script: testReadRow
Parâmetros: Idtpuser
Função: Comando para seleção no banco de dados

```
select * from tipousuario where
    <dtml-sqltest idtpuser type="int">
```

SCRIPT PARA ALTERAR O TIPO DE USUÁRIO NO MÓDULO REMOTO

```
from Products.PythonScripts.standard import url_quote
request = container.REQUEST
response = request.RESPONSE
form = request.form

try:
    if int(form.get('idtpuser', '-1')) >= 0:
        context.testUpdateRow(idtpuser=request.idtpuser,
usercriacao=request.usercriacao,desctpuser=request.desctpuser)
        destination = 'http://localhost:8080/Plone/paginas/sucesso'
        request.response.redirect(destination)
    else:
        destination = 'http://localhost:8080/Plone/paginas/erro'
        request.response.redirect(destination)
except:
    destination = 'http://localhost:8080/Plone/paginas/erro'
    request.response.redirect(destination)
```

Script: formSetup
Função: Carrega os dados na interface gráfica para serem alterados

```
request = container.REQUEST
form = request.form

if form.has_key('idtpuser') and not form.has_key('form.submitted') :
    res = context.testReadRow().dictionaries()
    if len(res) == 1:
        row = res[0]
        for key in row.keys():
            form[key] = row[key]
```

Script: testUpdateRow
Função: Comando para atualização no banco de dados

```
update tipousuario set
    <dtml-sqltest desctpuser type="string">,
    <dtml-sqltest usercriacao type="string">
where
    <dtml-sqltest idtpuser type="int">
```

Os *scripts* referentes aos cadastros e alterações das funcionalidades de tipos de usuários, grupos de usuários, tipos de operação, tipos de identificação, tipos de identificação de registros, tipos de mensagens e usuários são similares a estes, alterando-se os parâmetros e tabelas do banco

de dados a serem utilizadas.

SCRIPT PARA CONSULTAR O TIPO DE USUÁRIO

Script: Consulta

Função: Interface gráfica para receber o range de seleção inicial e final dos tipos de usuarios cadastrados no sistema

```
<html metal:use-macro="here/main_template/macros/master">
<head>
<title tal:content="resultado_da_pesquisa/title">The title</title>
</head>
<body><div metal:fill-slot="main">
<hr>
<h3>Consulta de Tipos de Usuários</h3>
<hr>
<form action="." method=get>
<p>Digite o Identificador do tipo de usuário inicial e final a pesquisar.</p>
<table>

<tr>
<td>Identificador Inicial:</td>
<td><input name=idtpuseri width="5" value="1"></td></tr>

<tr>
<td>Identificador Final:</td>
<td><input name=idtpuserf width="5" value="1"></td></tr>

<tr>
<td><input type=submit value="Pesquisar" name=Pesquisa:method></td>
</tr>

</table></form>
</div></body>
</html>
```

Script: Pesquisa

Função: Montar a interface gráfica de acordo com o pedido

```
<html metal:use-macro="here/main_template/macros/master">
<body metal:fill-slot="main" tal:define="results here/sql_consulta; start
request/start|python:0; batch python:modules['ZTUtils'].Batch(results, size=30,
start=start); previous python:batch.previous; next python:batch.next"><br>
<hr><h3>Resultado da Consulta</h3><hr><p>
<a tal:condition="previous" tal:attributes="href
string:${request/URL0}?start:string=${previous/first}"
href="previous_url">previous <span tal:replace="previous/length">30</span>
results</a>
<a tal:condition="next" tal:attributes="href
string:${request/URL0}?start:int=${next/first}" href="next_url">next <span
tal:replace="next/length">20</span>results</a></p>
<table border><tr>

<th>Identificador Usuário</th>
<th>Usuário de Criação</th>
<th>Data Criação</th>
```

```

<th>Descrição Tipo Usuário</th>

</tr>

<tal:x repeat="result batch" >
<tr>

<td><span tal:content="result/idtpuser"></span><input type="hidden"
name=idtpuser></td>
<td><span tal:content="result/usercriacao"></span></td>
<td><span tal:content="result/datacriacao"></span></td>
<td><span tal:content="result/desctpuser"></span></td>

</tr></tal:x>
</table><p>

<li id="document-action-voltar">
<a href="http://localhost:8080/Plone/consultas/Tipos de
Usuarios/Consulta">voltar</a>
</li>

<li id="document-action-print">
<a href="javascript:this.print();" title="">imprimir</a>
</li>

<div>
<form action="." method=post>
<br><table border><tr>
<hr><h3>Exclusão - Coloque as informações para exclusão:</h3><hr><br>

    <tr><th>Identificador Tipo Usuário</th>
    <td><input type="text" name="idtpuser" maxlength="5" size="5" value=""></td>
    <td><INPUT TYPE="submit" VALUE="excluir"
name="ScriptDelete:method"></td></tr>
</table></form></div><p>
</p></body></html>

```

Script: Resposta

Função: Exibir a resposta de exclusão com sucesso

```

<html metal:use-macro="here/main_template/macros/master">
<head>
<title tal:content="template/title">The title</title>
</head>
<body><div metal:fill-slot="main">
<h3><span tal:replace="options/mensagem | default">Exclusão com
sucesso!</span></h3>
<br>
<a href="http://localhost:8080/Plone/consultas/Tipos de
Usuarios/Consulta">voltar</a>
</div>
</body>
</html>

```

Script: Erro

Função: Exibir a resposta de erro na exclusão

```

<html metal:use-macro="here/main_template/macros/master">
<head>

```

```
<title tal:content="template/title">The title</title>
</head>
<body><div metal:fill-slot="main">
<h3><span tal:replace="options/mensagem | default">Dados não foram excluídos!
Verifique se os dados informados são corretos e se existe algum relacionamento
!</span></h3>
<br>
<a href="http://localhost:8080/Plone/consultas/Tipos de Usuarios"> <br>
voltar</a>
</div>
</body>
</html>
```

Os *scripts* referentes às consultas e exclusões das funcionalidades de tipos de usuários, grupos de usuários, tipos de operação, tipos de identificação, tipos de identificação de registros, tipos de mensagens e usuários são similares a esses, alterando-se os parâmetros e tabelas do banco de dados a serem utilizadas.