

AUTOMATIZAÇÃO DA GESTÃO DE ROTINAS DE MANUTENÇÃO E SUPORTE EM INFORMÁTICA

Carlos Henrique Jochims Spencer de Freitas¹

Luciano Beiestorf Rocha²

RESUMO

Este trabalho apresenta o desenvolvimento de um programa gerenciador de ordem de serviço voltado à manutenção e assistência técnica de computadores, além de ter a possibilidade de usar o programa em rede, podendo ser acessado em mais de uma máquina simultaneamente. Com o aumento das vendas em equipamentos na área da informática alavancando a prestação de serviço em manutenção, é necessário o auxílio de um programa que automatize e aumente a produtividade do profissional de TI que atua na manutenção destes equipamentos. O software foi implementado através do Lazarus, um Software Livre para o Desenvolvimento Rápido de Aplicações (RAD) gratuito com base no compilador FreePascal. Foi utilizado também vínculo ao banco de dados PostgreSQL e para administrá-la foi utilizado o pgAdmin que é uma interface administradora gratuita para este banco. Também foi utilizado o PgModeler, que é uma ferramenta de modelagem que auxilia em um conceito de diagrama entidade-relacionamento do banco de dados utilizado. Para o desenvolvimento do programa foram estudados outros programas semelhantes disponibilizados no mercado, considerando aspectos positivos de cada um, tornando assim, um programa fácil de utilizar e com uma interface amigável ao cliente, abrangendo todos os campos e informações que um técnico na área de TI precisa. Após a finalização, o programa foi implementado em um ambiente real, para avaliação de comportamento e testes.

Palavras-chave: software; manutenção; banco de dados; rede.

¹Estudante do Curso Técnico em Manutenção e Suporte em Informática do Instituto Federal de Educação, Ciência e Tecnologia Sul-rio-grandense, Câmpus Camaquã. E-mail: yroshima_rs@hotmail.com

²Professor Orientador do Instituto Federal de Educação, Ciência e Tecnologia Sul-rio-grandense, Câmpus Camaquã. Mestre em Engenharia Elétrica pela Pontifícia Universidade Católica do Rio Grande do Sul PUCRS, 2007 e Bacharel em Ciências da Computação pela Universidade Católica de Pelotas UCPEL, 1999. E-mail: luciano.rocha@camaqua.ifsul.edu.br

ABSTRACT

This study presents the development of a program that manages service orders concerning the maintenance and technical assistance of computers. The developed program may be used on network and may be accessed in more than one machine simultaneously. With the increase in sales related to computer equipments and the services related to this area, it is necessary a program that automate and increase the productivity of IT professionals that work on the maintenance of such equipments. The software was implemented through Lazarus which is an open source software to fast development of applications (RAD – Rapid Application Development) bases on the FreePascal compiler. A database PostgreSQL was used and to administrate it, a pgAdmin was chosen because it is a free interface administrator. A PgModeler was also used which is a modeling tool that assists in the concept of entity-relationship diagram of the database used. In order to develop the program, similar programs available on the market were studied. From this study, the positive aspects of these programs were kept used in the new program making it an easy program to use with a friendly interface to the client, covering all the fields and information that a technician in the IT area needs. Finally, the program was implemented in a real environment for evaluation and behavior test.

Key-words: software; maintenance; database; network.

INTRODUÇÃO

Nos últimos anos, a busca por equipamentos em informática vem crescendo a passos largos, cada vez mais as pessoas vem adquirindo computadores, notebooks, impressoras, entre outros, e consequentemente o aumento de serviço na área de manutenção em informática também vem crescendo. Com isto, o prestador de serviço necessita agilizar o atendimento a demanda, sendo necessário o auxílio de um programa que automatize e aumente sua produtividade.

Ao adquirir recentemente uma loja de informática, obtive uma grande dificuldade em encontrar um software que atendesse minhas necessidades para o controle de ordem de serviço, alguns programas eram extremamente precários em ambiente gráfico, outros eram ótimos em outros módulos, mas em ordem de serviço deixavam a desejar, outros tinham a estrutura do banco de dados muito precária, tornando difícil o seu manuseio. Tendo em vista este problema, foi proposto desenvolver um software, selecionando os aspectos positivos de cada um dos programas.

O objetivo deste trabalho consiste na criação de um software que controle ordens de serviço de uma maneira simples e clara, com uma interface amigável ao usuário, dando opções de cadastro de equipamento e também possibilitando a impressão de recibo de ordem de serviço.

1 REVISÃO DE LITERATURA OU REFERENCIAL TEÓRICO

1.1 PROGRAMA

Programa ou software, consiste em uma sentença escrita em uma linguagem computável, para a qual existe uma máquina capaz de interpretá-la. O software é composto por uma sequência de instruções (comandos) e declarações de dados, armazenável em meio digital. Ao interpretar o software, a máquina computável é direcionada à realização de tarefas especificamente planejadas, para as quais o software foi projetado. O programa é escrito em uma linguagem de programação, embora seja possível, com alguma dificuldade, escrevê-lo diretamente em linguagem de máquina. A programação é o processo de montagem deste software.

Segundo a própria Wikipédia, que descreve de um modo bem simples: “Programação é o processo de escrita, teste e manutenção de um programa de computador.”

1.2 FREE PASCAL

Conforme o site “<http://wiki.freepascal.org>”, Free Pascal é um compilador de código fonte aberto (open-source) com duas notáveis características: um grau elevado de compatibilidade com Delphi e disponibilidade em uma variedade de plataformas, incluindo Windows, Mac OS X, e Linux.

1.3 LAZARUS

O Lazarus é um Software Livre para o Desenvolvimento Rápido de Aplicações (RAD) com base no compilador Free Pascal. Com esta ferramenta é possível o desenvolvimento de aplicações multiplataforma. Isto significa que podemos escrever um código e a partir deste compilar aplicativos para executarem em Windows, Linux, Mac OS X ou outra plataforma compatível. Por isso, o lema do Lazarus de acordo com Patrick (2011) é “Escreva uma vez e compile em qualquer lugar”.

1.4 BANCO DE DADOS

Um banco de dados é um conjunto coerente e lógico de dados relacionados que possuem significância intrínseca. Segundo Antunes (2008, p.3), “estes dados representam aspectos do mundo real e devem ser mantidos para atender aos requisitos necessários.” Para gerenciar um banco de dados é usado um Sistema Gerenciador de Bancos de Dados (SGBD), que é um software construído para facilitar as atividades de definição, construção e manipulação de bancos de dados. Exemplos de SGBD são: PostgreSQL, Firebird, SQLServer, MySQL, entre outros.

1.5 POSTGRESQL

Conforme o próprio site do desenvolvedor “<http://www.postgresql.org/about/>”, PostgreSQL é um sistema gerenciador de banco de dados open source poderoso, pois além de ser gratuito, ele possui uma confiabilidade, estabilidade e segurança na estrutura do banco de dados.

1.6 PGMODELER

Para auxílio à criação e administração de tabelas e colunas do banco de dados, existem programas que auxiliam a modelar e visualizar a interação e criação da estrutura do banco de dados, PgModeler é uma ferramenta de código aberto para bancos de dados de modelagem que funde os conceitos clássicos de diagramas entidade-relacionamento com características específicas que só PostgreSQL implementa.

2 METODOLOGIA

2.1 ANÁLISE DE MERCADO

Como a proposta deste trabalho já existe, antes de começar o desenvolvimento do programa, primeiramente foi feita uma análise dos softwares disponíveis no mercado que possuísem controle de ordem de serviço. Esse estudo foi feito com base nos programas:

- TagComércio, desenvolvida pela TagSoft, situada em Belo Horizonte/MG, site: www.tagsoft.com.br;

- OS Geral, desenvolvida pela FpqSystem, situada em Alvorada/RS, site www.fpqsystem.com.br;
- Loja de Informática, desenvolvida pela Pazello Informática, site: www.epratico.com.br;

Todos estes programas possuíam outros complementos como ponto de vendas, controle de estoque, financeiro, mas somente foi avaliado o controle de ordem de serviço. O programa TagComercio, possuía um ótimo programa, com um visual bem estruturado, e completo, mas nas ordens de serviço, o programa deixava a desejar, dando liberdade apenas de situação do serviço e observações, os serviços realizados tinham de ser previamente cadastrados como um produto, dificultando a utilização. Já no software Loja de Informática foi totalmente o inverso, o programa possuía um visual muito ruim, de difícil manuseio, apresentando algumas opções a mais no controle dos serviços prestados, mas apresentando muita instabilidade. Por último, o programa da FpqSystem foi o que teve a melhor estrutura para ordens de serviço, mas infelizmente ficou devendo na composição do banco de dados, ocasionando problemas ao preencher campos como caixas de texto, por exemplo, que ao clicá-las, o programa começava a digitar a partir de onde foi clicado, e não no início da linha, nem da caixa de texto.

2.2 PRÉ-DESENVOLVIMENTO

Através do estudo dos programas existentes no mercado, foram considerados os pontos positivos de cada um, para que pudéssemos aproveitá-los para o desenvolvimento do programa. Para produção do software, foram também utilizados os conhecimentos dos programas estudados no decorrer do curso.

O Lazarus (Figura 01), um compilador FreePascal, foi o programa escolhido para a criação do software. Ele pode ser adquirido fazendo o download no site “<http://www.lazarus.freepascal.org/>”, a versão utilizada para o desenvolvimento do trabalho foi a 1.2.2.

Como o software a ser desenvolvido tem um sincronismo com serviço de banco de dados, decidiu-se por utilizar o PostgreSQL, versão 9.3, que pode ser adquirido através do site “<http://www.postgresql.org/>”. Para nos auxiliar na criação e visualização de entidade-relacionamento entre as tabelas do banco, descobrimos um programa desenvolvido por um brasileiro, chamado pgModeler (Figura 02), que é uma ferramenta de modelagem de dados criada especificamente para o PostgreSQL e está em fase Alpha. Quando descobriu-se este programa ele era de código aberto e gratuito, mas após dois meses de sua utilização, o

desenvolvedor começou a vender o software, então utilizamos a versão 0.7.0, que foi a última atualização gratuita, atualizações posteriores devem ser compradas para uso e podem ser adquiridas no site <http://www.pgmodeler.com.br/>.

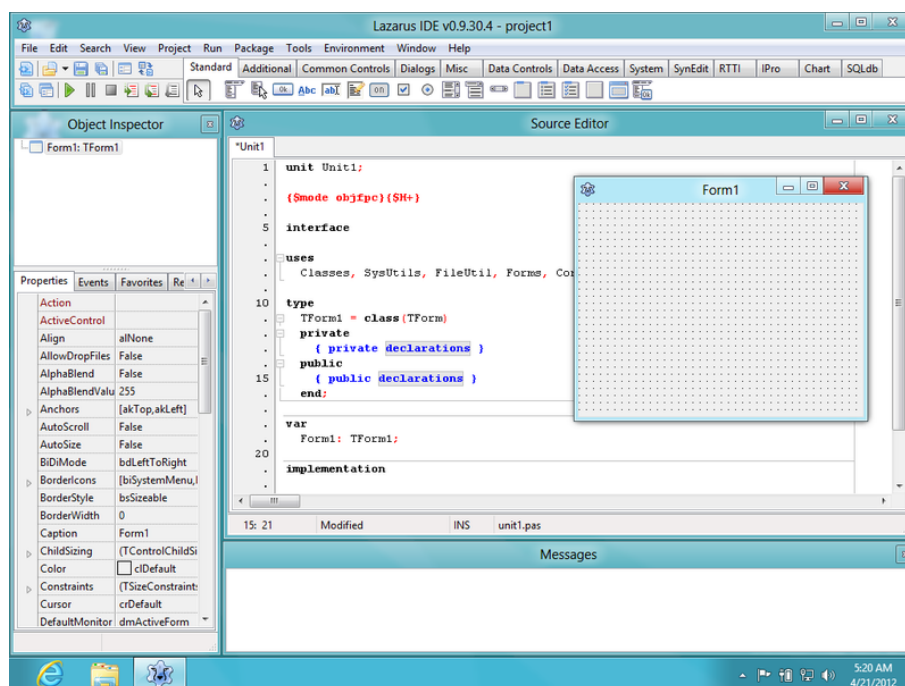


Figura 01 – Tela do inicial do programa Lazarus – Fonte: <http://wiki.lazarus.freepascal.org/Screenshots>.

2.3 O PROGRAMA

Primeiramente foi produzido um esboço com o qual compôs-se o que iria ser trabalhado e como seria feita a relação entre estes campos, a seguir foi desenvolvida uma modelagem com o programa pgModeler, ao mesmo tempo em que eram construídas as tabelas com as colunas, onde podiam ser vistas a relação entre as elas (Figura 02).

Com o início da criação das colunas e das tabelas, tendo as primeiras informações, iniciou-se a criação visual do programa proposto. O programa possui uma tela inicial (Figura 03), nela se tem acesso às principais informações do programa (Figura 04), como lista de clientes, funcionários, ordem de serviço, ferramentas do sistema, entre outros, trabalhando como abas, tornando o programa mais rápido e com menos linhas de comando, ao invés de criar janelas para cada opção.

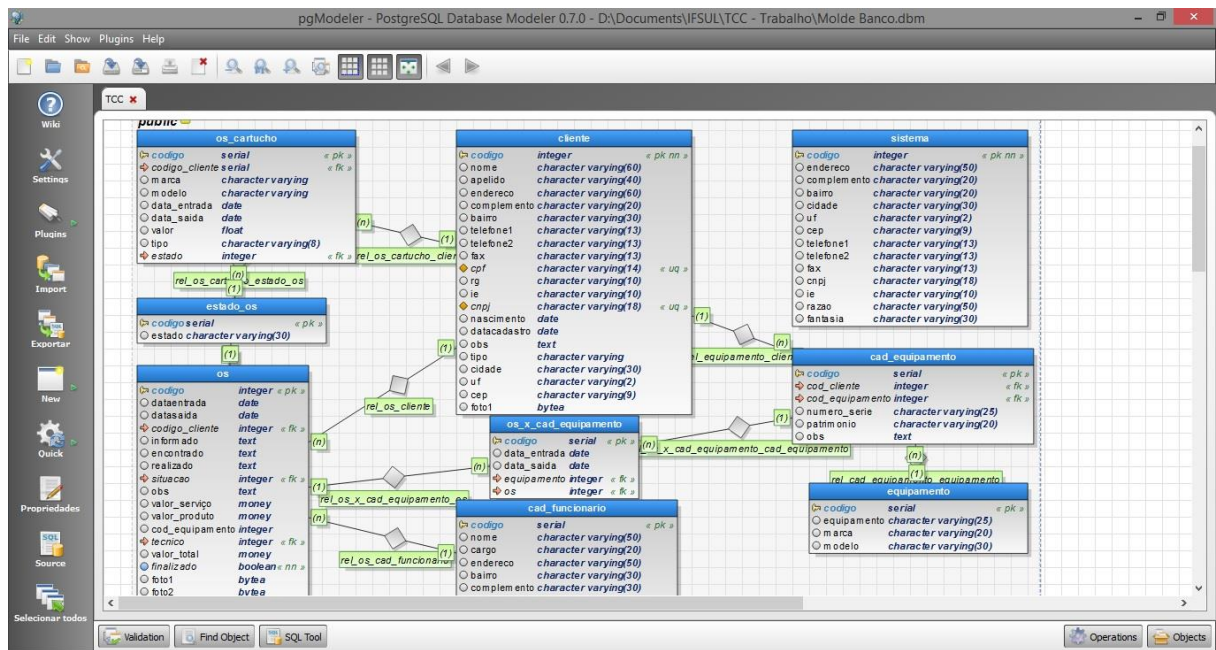


Figura 02 – Visualização do diagrama entidade-relacionamento do banco de dados no programa pgModeler.



Figura 03 – Imagem da tela inicial do programa

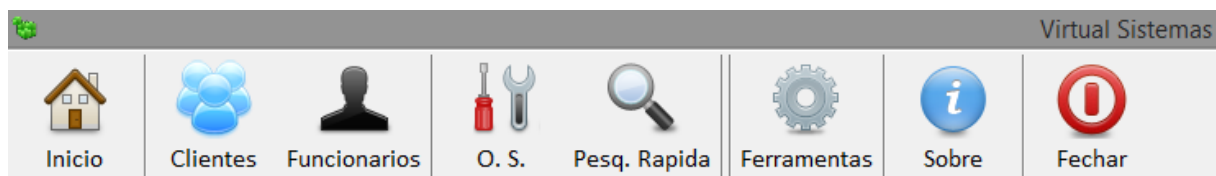


Figura 04 – Imagem do cabeçalho do programa.

2.3.1 CLIENTES

No menu “Clientes” (Figura 05) podemos visualizar a lista de clientes já cadastrados no sistema. Na parte esquerda temos as opções:

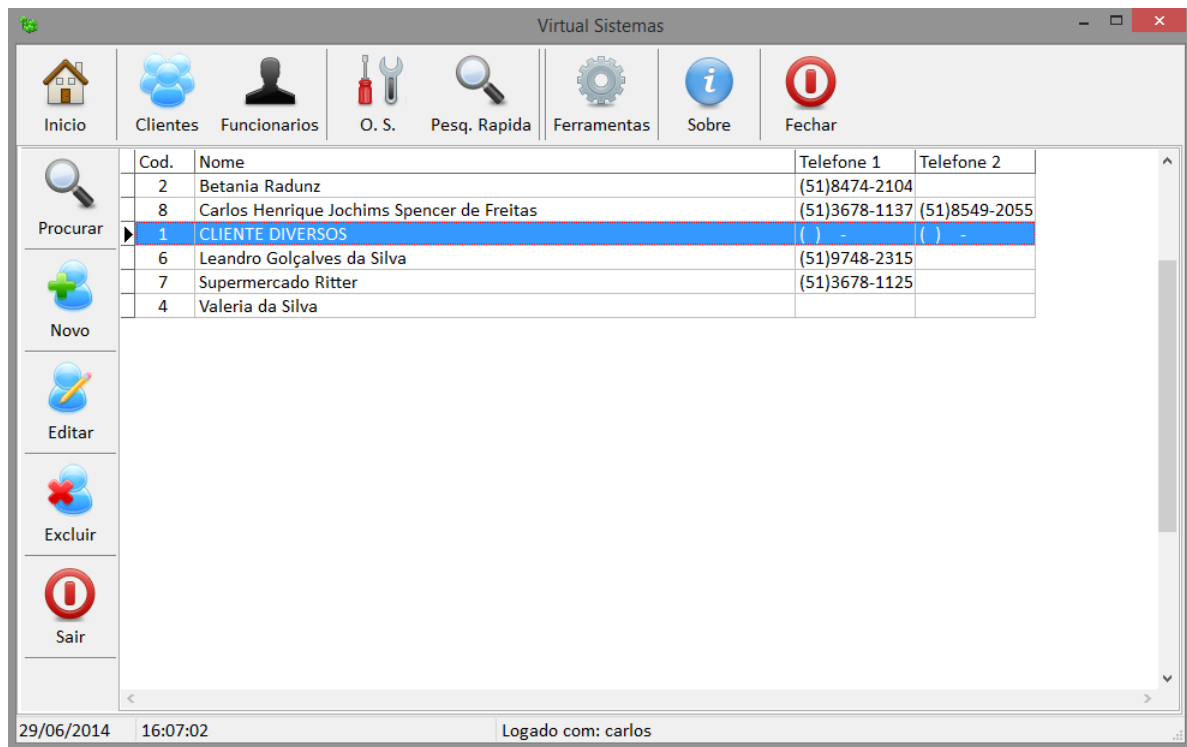


Figura 05 – Imagem da lista de clientes.

- Procurar: realiza uma pesquisa através do nome do cliente;
- Novo: abrirá uma janela de novo cadastro para adicionar um novo cliente (Figura 06);
- Editar: irá abrir uma janela para editar o cadastro de cliente já existente (Figura 06);
- Excluir: Irá deletar cadastro de cliente selecionado previamente;
- Sair: retornará a tela inicial do sistema;

The image shows a software window titled "Cliente" with a standard Windows-style title bar (minimize, maximize, close buttons). The form is organized as follows:

- Top Section:**
 - Tipo de Cliente:** Two radio buttons, "Física" (selected) and "Jurídica".
 - Código:** A text input field.
 - Data Cadastro:** A date input field showing "29/06/2014".
 - Buttons:** A green "Salvar" button with a checkmark icon and a red "Cancelar" button with an 'X' icon.
- Form Fields:**
 - Nome:** A text input field.
 - Apelido:** A text input field.
 - Endereço:** A text input field.
 - Complemento:** A text input field.
 - Bairro:** A text input field.
 - Cidade:** A text input field containing "Cristal".
 - CEP:** A text input field containing "96195-000".
 - UF:** A dropdown menu showing "RS" with a downward arrow.
 - Telefone 1:** A text input field.
 - Telefone 2:** A text input field.
 - FAX:** A text input field.
 - RG:** A text input field.
 - CPF:** A text input field.
 - Nascimento:** A text input field.
 - I.E.:** A text input field.
 - CNPJ:** A text input field.
- Observações:** A large text area at the bottom for additional notes.

Figura 06 – Imagem do cadastro de novo cliente e edição.

2.3.2 FUNCIONÁRIOS

A opção “Funcionários” (Figura 07) lista o cadastro dos mesmos, assim como na opção de “Clientes”, tendo as mesmas opções a esquerda do item anteriormente citado. No cadastro de funcionário têm-se as principais informações pessoais do mesmo, e um diferencial, a possibilidade de poder cadastrar uma senha de acesso ao sistema, que poderá ser utilizada posteriormente, tendo em vista que este possui um sistema de segurança de login ao abrir o programa (Figura 08).

Figura 07 – Imagem do cadastro de funcionário.

Figura 08 – Tela de login do sistema.

2.3.3 ORDEM DE SERVIÇO

Esta opção é o foco do trabalho, a tela “Ordem de Serviço” (Figura 09), mostrando a lista de OS cadastradas no sistema. Nesta tela, podemos criar, editar e cancelar as ordens, no centro, em grade, temos algumas informações sobre elas, como número de OS, cliente, telefone, situação e valor.

Ao clicar para criar ou editar uma ordem de serviço, abre-se uma nova janela (Figura 10) onde têm-se as informações como nome do cliente e seus telefones, além de um botão de pesquisa de clientes para associarmos ao serviço, situação (andamento do serviço), data da abertura e fechamento da ordem, a direita temos campos para colocar o valor gasto em peças e a mão de obra da OS, assim como o valor total do serviço, com a possibilidade de salvar as informações editadas, finalizar (quando for entregar o equipamento ao cliente), e a opção imprimir, que gera um comprovante da ordem de serviço. Para a impressão foi adicionado um

componente desenvolvido por uma empresa brasileira chamado Fortes Reporte, que gera o modelo a ser impresso.

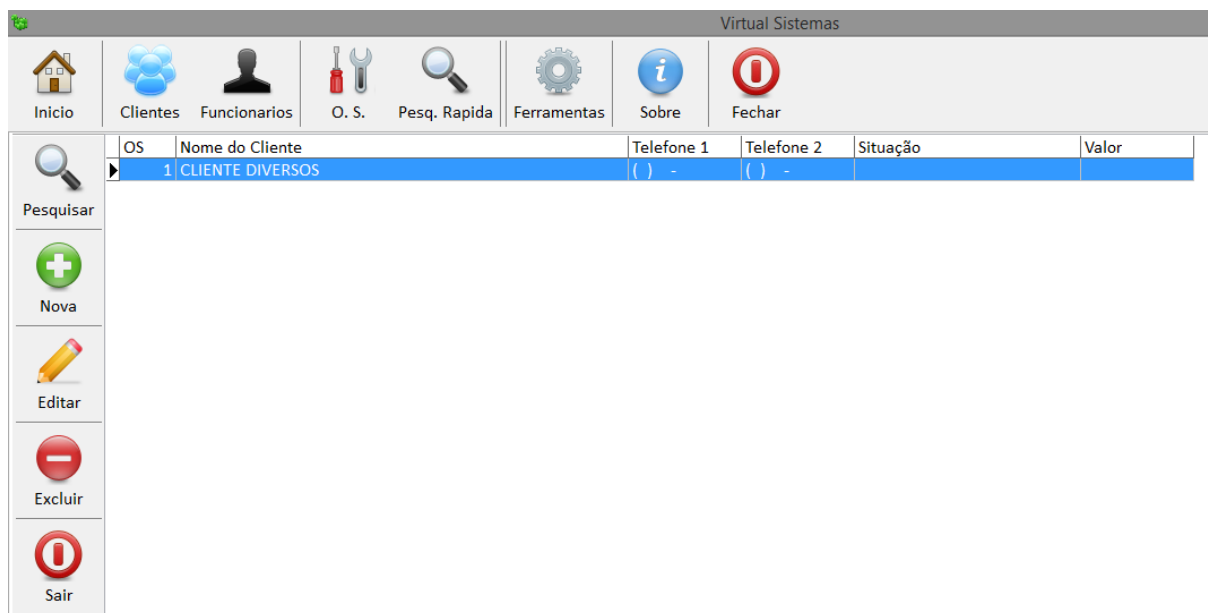


Figura 09 – Tela de controle de “Ordem de Serviço”.

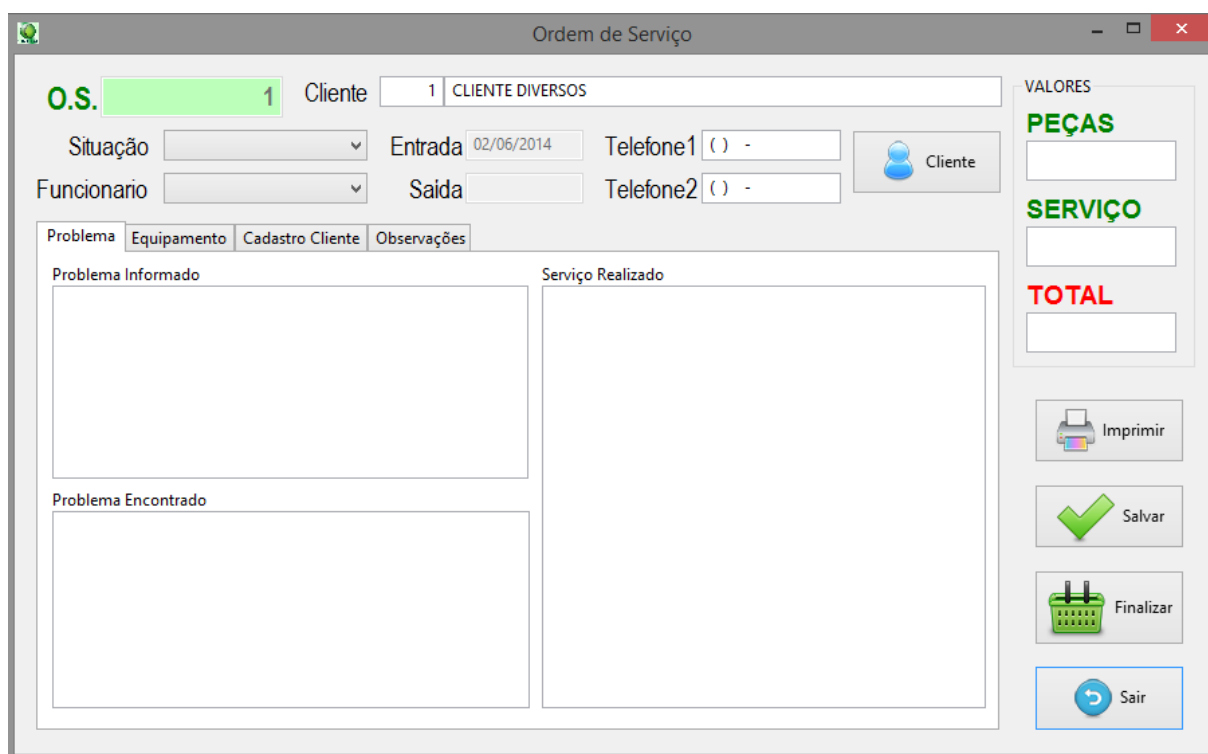


Figura 10 – Tela de abertura ou edição de “Ordem de Serviço”.

Ao centro temos abas onde a primeira, “Problema” são os problemas informados, encontrados e realizados no serviço. Na aba “Equipamento” (Figura 11) podemos colocar

dados referente ao equipamento, como tipo (notebook, computador, estabilizador, monitor, etc.), marca, modelo, número de série, acessórios, e outros.

The screenshot shows the 'Equipamento' tab of a software interface. At the top, there are tabs for 'Problema', 'Equipamento', 'Cadastro Cliente', and 'Observações'. The 'Equipamento' tab is active. It contains several input fields: 'Tipo de Equipamento', 'Marca', 'Modelo', 'Numero de Série / Patrimonio', and 'Acessórios'. Below these fields are two large rectangular areas labeled 'Foto 1' and 'Foto 2', each with a small 'Foto 1' or 'Foto 2' button above it. On the right side of the interface, there is a vertical sidebar with icons for 'SERV', 'TOT', and a calculator icon.

Figura 11 – Tela da aba “Equipamento” em “Ordem de Serviço”.

Assim como a aba “Cadastro Cliente” (Figura 12), ela mostra o restante dos dados cadastrais do cliente. E a última aba serve para observações complementares da ordem de serviço.

The screenshot shows the 'Cadastro Cliente' tab of the same software interface. It contains various input fields for client information: 'Tipo de Cliente' with radio buttons for 'Física' (selected) and 'Jurídica'; 'Nascimento'; 'Codigo' with a dropdown menu showing '1'; 'Data Cadastro' with a date field showing '21/05/2014'; 'Endereço'; 'Complemento'; 'Bairro'; 'Cidade' with a dropdown menu showing 'Cristal'; 'CEP' with a field showing '96195-000'; 'UF' with a dropdown menu showing 'RS'; 'RG'; 'CPF'; 'I.E.'; and 'CNPJ'. On the right side, the same vertical sidebar is visible.

Figura 12 – Tela da aba “Cadastro Clientes” em “Ordem de Serviço”.

2.3.4 FERRAMENTAS

Na opção “Ferramentas” (Figura13) temos acesso a janela de “Dados da Empresa” a qual cadastra as informações da empresa administradora do programa (Figura 14). A esquerda temos acesso a configurações do programa para acesso em rede, possibilitando fazer

configurações de servidor e cliente. A hospedagem do banco de dados pode ser indiferente do serviço do programa, ou seja, podemos ter um servidor (Linux ou Windows) rodando o serviço PostgreSQL, mesmo não havendo o software criado no mesmo equipamento, pois o banco não é ligado a ele, e demais máquinas com o programa sendo utilizado como terminais puxando o banco do servidor através da rede, aumentando a segurança das informações armazenadas trabalhando na porta de comunicação 5432.

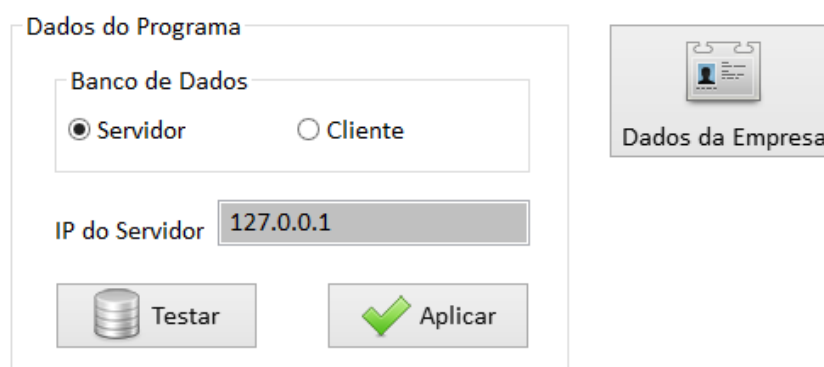


Figura 13 – Tela da opção ferramentas

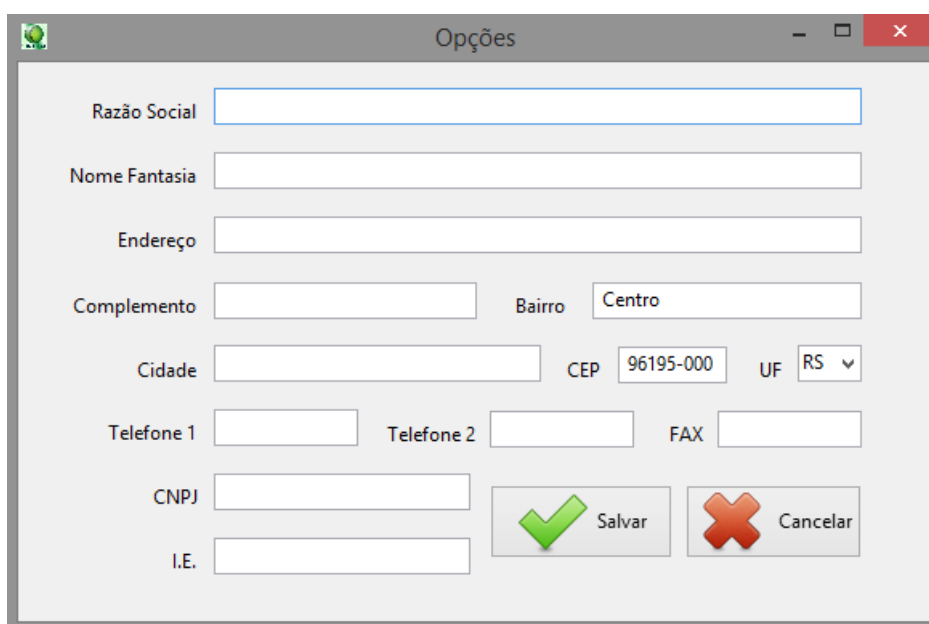


Figura 14 – Tela do cadastro da empresa administradora do programa

3 CONCLUSÃO

O software foi implementado em um ambiente real durante um dia, e teve dados reais de clientes cadastrados e entrada de equipamentos também. O programa atendeu bem as expectativas que deveriam ser realizadas, mas infelizmente não pode abolir o outro já

existente por não possuir mais módulos como controle de estoque, financeiro e ponto de venda, mas o serviço ao qual ele presta foi bem apreciado. Ele também apresentou algumas dificuldades para acessar o banco através da rede, mas estes consistem em alguns problemas de permissão.

O programa tem como foco a ordem de serviço, mas foi elaborado de forma que podem ser adicionados outros módulos posteriormente, como financeiro, ponto de vendas e controle de estoque, ou seja, futuramente pode ser implementada uma continuação do trabalho.

REFERÊNCIAS

PATRICK, Jean. “**Desenvolvendo Aplicativos com Lazarus**”. Ano 2011. <https://agbook.com.br/book/46758--Desenvolvendo_Aplicativos_com_Lazarus> Acesso em 24 de Março de 2014.

PASCAL, Wikipédia <<http://wiki.freepascal.org>> Acesso em Maio de 2014.

PGMODELER, Wikipédia <<http://www.pgmodeler.com.br/wiki/doku.php?id=start>> Acesso em Abril de 2014.

ANTUNES, Rubens. “**Delphi Faça uma Aplicação Comercial**”. Ano 2008. Editora Ciência Moderna.

TALMA, Thiago Moreira. “**Desenvolvimento de software de auxílio ao fluxo e ao compartilhamento de informações administrativas em ambientes empresariais**”. 2006. <http://www.ufjf.br/ep/files/2009/06/tcc_fev2006_thiagomoreiratalma.pdf>. Acessado em 03 de Junho de 2014.