

7. SISTEMA DE GERAÇÃO DE ORDENS DE SERVIÇOS

Evandro Roson dos Santos

7.1. Introdução

Foi solicitado pela docente das disciplinas de Linguagem de programação IV e Banco de Dados um sistema Web em PHP que conseguisse fazer alterações em pelo menos uma tabela do Banco de Dados MySQL, com a inclusão de XML, JSON, Session e um diferencial escolhido pelo discente.

O sistema tem como objetivo auxiliar os atendentes de qualquer prestadora de serviços, para que seja preenchida as ordens de serviço digitalmente com facilidade. Os técnicos utilizam desta ordem gerada para realizar o serviço proposto e o cliente uma cópia deste arquivo como comprovante via e-mail. O administrador da empresa cadastra os atendentes, para que eles possam interagir com o sistema e gerar as ordens de serviço.

7.2. Projeto do Banco de Dados

Um sistema para uma empresa de informática gera ordens de serviço para a organização e o cliente.

A organização possui dois tipos de funcionário(cod, nome, telefone, endereço (logradouro, número, cep, bairro)): os atendentes(nº crachá, senha) e os técnicos(especialidade).

O administrador do sistema (cod, senha, email, nome) irá alterar permissões(adicionar, excluir, modificar) dos atendentes(data, hora, alteração) que possuem permissão de gerar as ordens de serviço.

Os atendentes recebem os clientes(telefones*, nome, endereço(logradouro, número, cep, bairro), cpf) e anotam a data e hora.

Com a recepção do cliente pelo atendente é gerado a ordem de serviço((nº, nome, endereço(logradouro, número, cep, bairro)telefone, email, tipo de máquina, marca da máquina, modelo da máquina, descrição do serviço, valor do serviço, desconto do serviço, nº de parcelas) com todas as informações descritas pelos clientes no preenchimento, marcando a data e hora.

A ordem de serviço gerada na recepção é recebida pelos técnicos para que o serviço seja realizado.

A ordem de serviço possui serviços (codigo, servico, descrição).

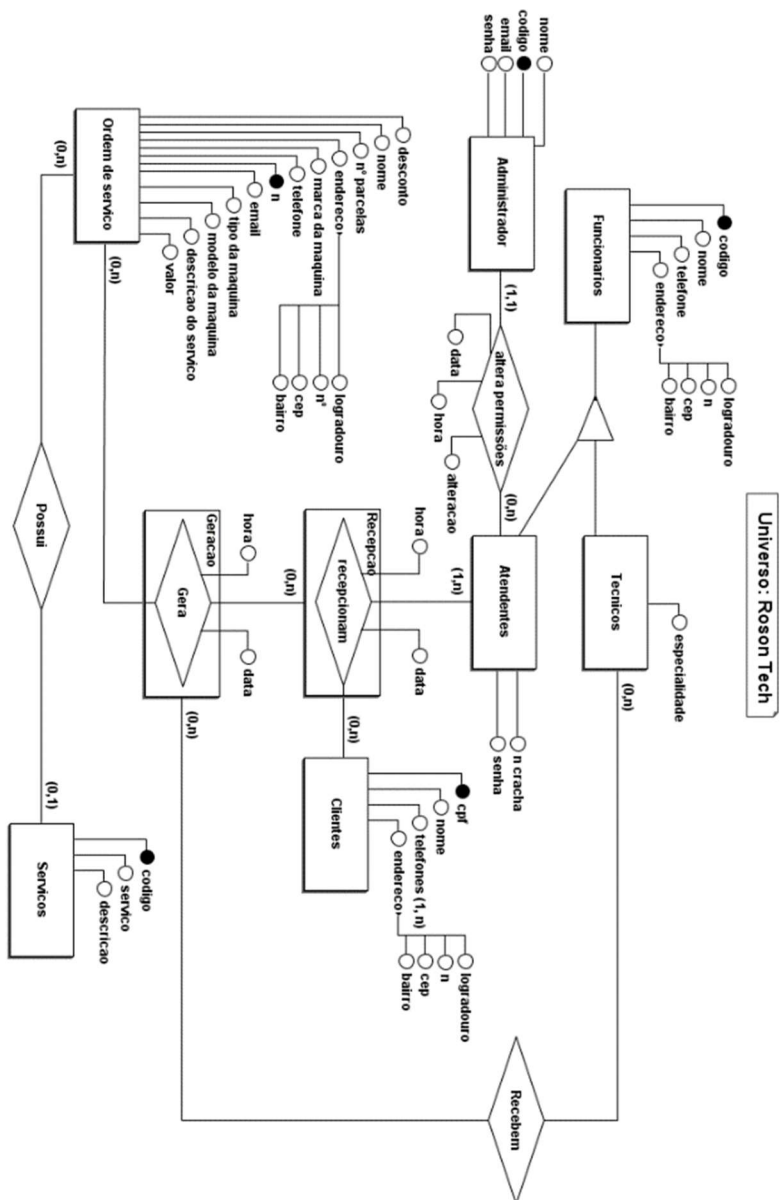


Figura.7.1 DE-R do Website

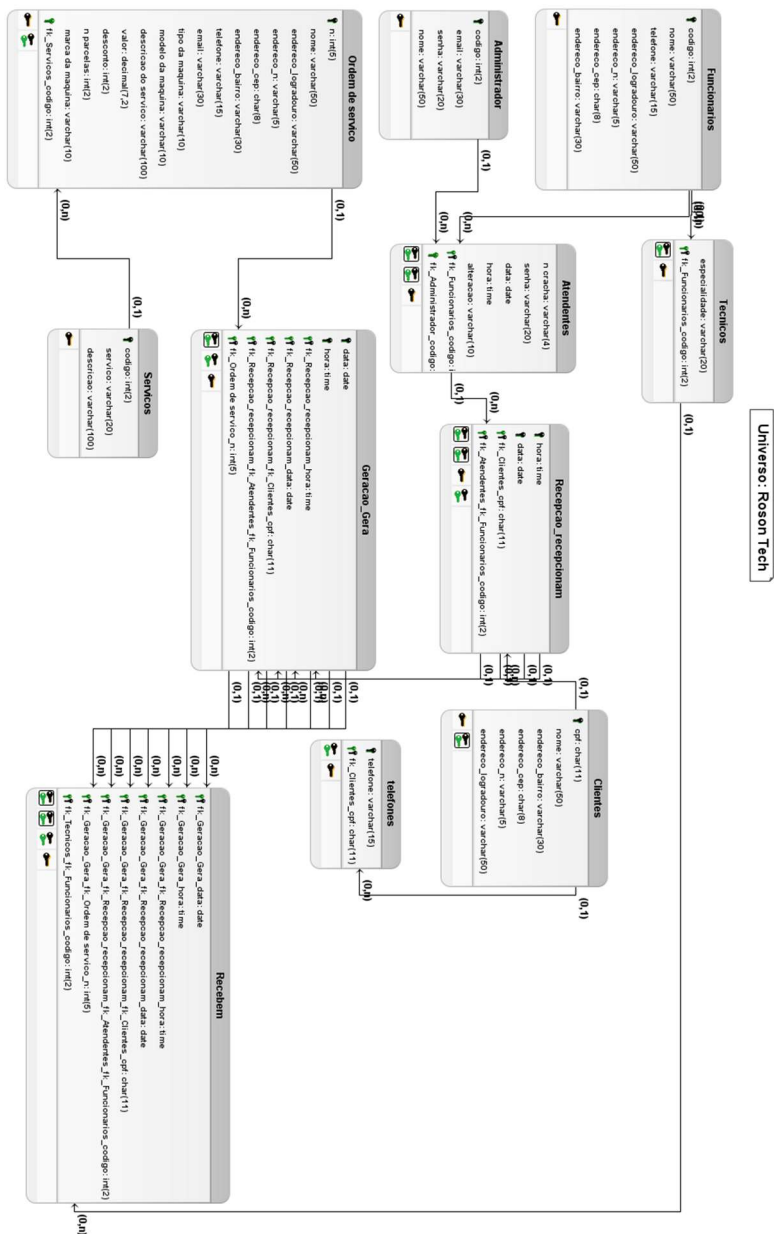


Figura.7.2 MR do Website

7.3. Desenvolvimento

O sistema de geração de ordens de serviços está dividido em 02 módulos principais:

- **Administrador:** onde é alterado as permissões dos funcionários que podem gerar as ordens de serviço.
- **Usuário:** Onde os funcionários da instituição podem preencher a ordem de serviço no momento da recepção do cliente.



Figura.7.3 Administrador

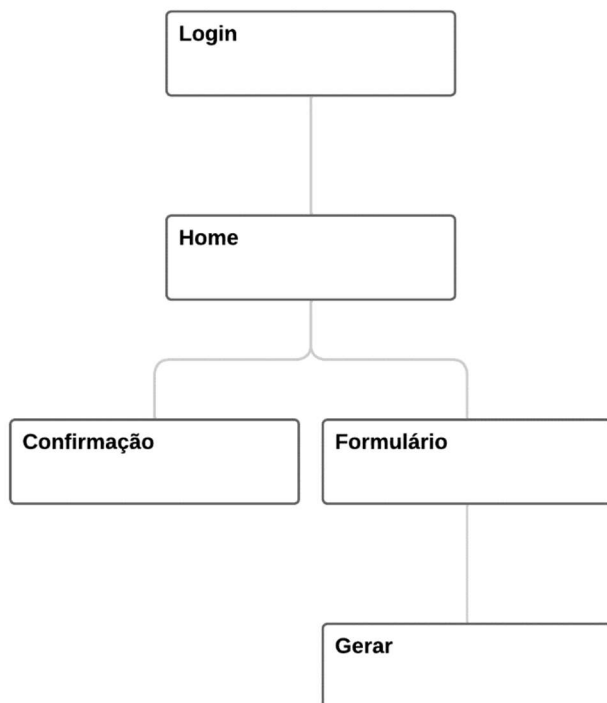
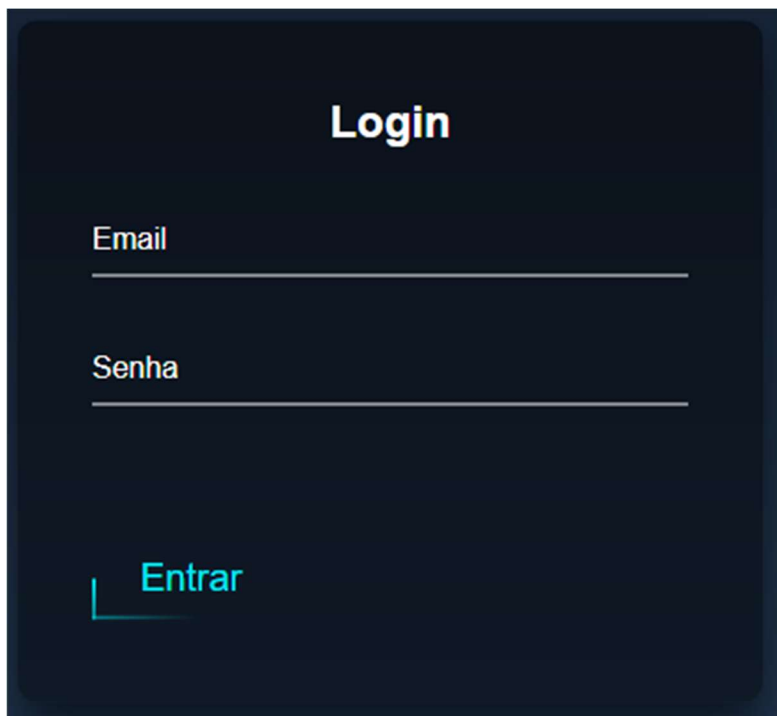


Figura.7.4 Usuário

Administrador

Somente pela tela de login (mostrado na figura 7.5) é possível o administrador entrar no sistema.



The image shows a login interface with a dark blue background. At the top, the word "Login" is displayed in white. Below it, there are two input fields: "Email" and "Senha", both with white text and white borders. At the bottom left, there is a button labeled "Entrar" in white text, with a white border and a small white icon to its left.

Figura.7.5 Login

Na página “Managment” (figura 7.6) é possível incluir, alterar ou excluir o cadastro dos atendentes que estão aptos a utilizar o sistema para gerar ordens de serviço e sair do sistema.

Roson Tech Management

Encerrar

Nº Crachá

Senha

Adicionar



ID	Nº Crachá	Senha	Modificação	Data	Hora	Alterar	Excluir
19	1475	123456	Incluir	2022-11-13	23:07:35		

Figura.7.6 Management

Na página de confirmação (figura 7.7) é onde acontece a validação da ação tomada pelo administrador na página de “Management”.

Atenção!

Tem certeza que deseja encerrar a sessão?

Cancelar

Confirmar

Figura.7.7 Confirmação

Usuário

A página de login para o usuário (atendentes da empresa) possui o mesmo funcionamento da página de login do administrador, como é mostrado na figura 7.5. Assim como a página de confirmação, mostrado na figura 7.7.

A “Home” possui funcionalidades, como visualizar os fornecedores da organização, sair do sistema e escolher qual o tipo de serviço escolhido pelo cliente para o preenchimento posterior da ordem de serviço, como é mostrado na figura 7.8.



Figura.7.8 Home

Após a escolha do serviço, na página de “Formulário” é preenchido todos os dados do cliente em relação ao serviço desejado, como mostrado na figura 7.9.

Ordem de Serviço N°11

Serviço: Formatação Simples

Dados do Cliente

Nome Completo*: _____ Data de Recebimento: Jundiaí, 19 de Novembro de 2022

Logradouro*: _____ Nº*: _____

Telefone*: _____ Email*: _____

Dados da Máquina

Tipo*: Desktop ▼ Marca*: _____ Modelo*: _____

Descrição do Serviço*: _____

Figura.7.9 Formulário

Quando o formulário é preenchido, o site é redirecionado para a página “Gerar”, onde é realmente gravado no sistema os dados da ordem de serviço e enviado para os clientes (via e-mail) e técnicos. Também é possível retornar a “Home”, como é mostrado na figura 8.0.



Figura.8.0 Gerar

7.4. Diferencial

O sistema envia e-mail como comprovante para o cliente ficar a par de suas informações divulgadas à organização.

```

1  <?php
2  use PHPMailer\PHPMailer\PHPMailer;
3  use PHPMailer\PHPMailer\SMTP;
4  use PHPMailer\PHPMailer\Exception;
5  require 'vendor/autoload.php';
6  $mail = new PHPMailer(true);
7  try {
8      $mail->SMTPDebug = SMTP::DEBUG_SERVER;
9      $mail->isSMTP();
10     $mail->Host      = 'smtp.gmail.com';
11     $mail->SMTPAuth   = true;
12     $mail->Username   = 'seuemail@gmail.com';
13     $mail->Password   = 'suasenha';
14     $mail->SMTPSecure = PHPMailer::ENCRYPTION_SMTPS;
15     $mail->Port       = 465;
16     $mail->setFrom('exemplo@gmail.com', 'Exemplo');
17     $mail->addAddress('exemplo@yahoo.com.br', 'Exemplo');
18     $mail->addAttachment('imagem.png');
19     $mail->isHTML(true);
20     $mail->Subject    = 'Assunto';
21     $mail->Body       = 'Corpo';
22     $mail->AltBody    = 'Corpo sem formatação';
23     $mail->send();
24     echo 'Mensagem enviada com sucesso';
25 } catch (Exception $e) {
26     echo "Mensagem não enviada. Erro: {$mail->ErrorInfo}";
27 }
28 ?>

```

Figura.8.1 Código fonte envio de Email

Primeiramente, antes de começar a digitar o código mostrado na figura 8.1, deve-se fazer o download das bibliotecas necessárias do repositório oficial do Github (<https://github.com/PHPMailer/PHPMailer>), onde é possível visualizar toda a documentação para o envio de e-mail. No caso deste sistema, as bibliotecas foram instaladas pelo Composer (gerenciador de dependências do PHP), digitando “composer require phpmailer/phpmailer” na pasta desejada

para a instalação, mas existem outras formas de realizar a instalação das bibliotecas, todas descritas no repositório.

Começamos o código importando as classes PHPMailer, como mostrado nas linhas 2, 3 e 4.

Na linha 5 é carregado o autoloader do Composer.

Após, deve-se criar uma instância que será usada para setar todas as informações necessárias para o envio do e-mail, passar True como parâmetro habilitam as exceções.

Dentro do Try Catch, é setada todas as informações para que, caso ocorra algo inesperado, se possa tratar esse erro:

Servidor:

- **SMTPDebug: = SMTP::DEBUG_SERVER;** Esse parâmetro é opcional e mostra todas as informações de Debug do processo de envio.
- **isSMTP();** Função destinada a setar o envio usando SMTP (protocolo de transferência de email simples, que define a padronização das informações que identificam cada email e o caminho que ele deve percorrer para ser entregue de forma íntegra, sigilosa e segura), que é um protocolo muito utilizado pelos sistemas de e-mail hoje em dia.
- **Host;** Define qual servidor SMTP será usado.
- **SMTPAuth = true;** Habilita a autenticação SMTP.
- **Username;** Define o usuário do servidor SMTP.
- **Password;** Define a senha do servidor SMTP.

- **SMTPSecure** = **PHPMailer::ENCRYPTION_SMTPS;** Ativa criptografia TLS implícita.
- **Port;** Define a porta TCP para se conectar. A porta 465 é uma das portas usadas para submissão de SMTP.

Destinatários:

- **setFrom;** Define qual e-mail será utilizado para enviar.
- **addAddress;** Define quem irá receber o email, o endereço é obrigatório e o nome opcional. Podem ser adicionados mais de um destinatário.

Anexos:

- **addAttachment;** Define anexos para serem enviados, basta colocar o caminho do arquivo. Podem ser adicionados mais de um anexo.

Conteúdo:

- **isHTML(true);** Define que o conteúdo a ser enviado vai ter formatação HTML.
- **Subject;** Define o assunto do e-mail.
- **Body;** Define a mensagem a ser enviada, podendo colocar tags HTML.
- **AltBody;** Alguns e-mails não suportam tags HTML, então aqui é definido o mesmo texto

do body, mas sem formatação, para não existir o risco de o destinatário receber a informação ilegível.

Envio:

- **send();** Função que realiza o envio do e-mail com todas as informações setadas.

7.5. Considerações Finais

O Web site desenvolvido, apesar de simples e incompleto por fazer parte de um trabalho acadêmico, teve todos os objetivos e requisitos solicitados pela docente do curso cumpridos.

Nada impede que, futuramente, o site seja completado com a utilização de todas as tabelas presentes no banco de dados e definições de layouts com maior nível de padronização e interatividade, para que seja utilizado em organizações que necessitam da anotação de dados de clientes que à visitam, automatizando este processo.

Por ser o único desenvolvedor deste projeto, existiram muitos desafios que foram surgindo no decorrer do desenvolvimento. Isso teve como resultado, um grande acréscimo no conhecimento, me preparando para novos desafios. Por fim, todo o processo foi satisfatório e muito divertido.

7.6. Bibliografia

Marcus Bointon. **PHPMailerGenericName**. Disponível em:
<<https://github.com/PHPMailer>>. Acesso em: 03 out. 2022.