



**UNIVERSIDADE PAULISTA**

**BRUNA LINO MOYANO – RA G095057**

**JONATHAN SOUZA MOYANO – RA G081960**

**LUIZ FELIPE LUZ SANTOS – RA N029HA7**

**GUILHERME ALMEIDA JUSTINO –RA G0725D0**

**HELOISA DOS SANTOS COSTA RA - N029447**

**PROJETO INTEGRADO MULTIDISCIPLINAR**

**Projeto de Gerenciamento de uma Startup**

**UNIP (SANTOS)**

**2022**

**BRUNA LINO MOYANO – RA G095057**

**JONATHAN SOUZA MOYANO – RA G081960**

**LUIZ FELIPE LUZ SANTOS – RA N029HA7**

**GUILHERME ALMEIDA JUSTINO –RA G0725D0**

**HELOISA DOS SANTOS COSTA RA - N029447**

## **PROJETO INTEGRADO MULTIDISCIPLINAR**

### **Projeto de Gerenciamento de uma Startup**

Projeto Integrado Multidisciplinar – PIM  
apresentado a Universidade Paulista – UNIP, para  
avaliação semestral no curso de Análise e  
Desenvolvimento de Sistemas.

Orientador: Professor João Carlos Martins

**UNIP (SANTOS)**

**2022**

## RESUMO

O projeto consiste em apresentar um software no qual administre as funções diárias, tais como principais cadastros de clientes e funcionários, orçamentos, ordem de serviços, fluxo administrativo e operacional de gastos, receitas e relatórios financeiros para melhor a comunicação, rendimento e agilidade entre os colaboradores de uma Startup.

A Startup fictícia que desenvolvemos foi apelidada de Ctech, no qual oferece serviços de manutenção e prevenção em notebooks com avaliação e prazo de 24 horas para resolução dos problemas.

A necessidade da criação deste software se dá pela expansão da empresa, agora com mais funcionários, recursos e investimento.

**Palavras-chave:** Startup, software, investimento.

## **ABSTRACT**

The project consists of presenting a software in which to manage daily functions, such as main customer and employee records, budgets, order of services, administrative and operational flow of expenses and revenues and financial reports to improve communication, performance and agility between Startup employees.

The fictitious Startup we developed was dubbed Ctech, which offers maintenance and prevention services for notebooks with evaluation and a 24-hour deadline for problem resolution.

The need to create this software is due to the company's expansion, now with more employees, resources and investment.

**Keywords:** Startup, software, investment.

## SUMÁRIO

1	INTRODUÇÃO .....	5
1.1	O QUE É STARTUP? .....	5
2	DESENVOLVIMENTO .....	6
2.1	SOBRE A STARTUP .....	6
3	FUNCIONARIOS .....	7
3.1	GERENTES .....	7
3.2	TÉCNICOS .....	7
3.3	RECEPCIONISTA .....	8
4	PROSPECÇÃO FINANCEIRA .....	10
5	REDES .....	11
5.1	CONCEITO DE TOPOLOGIA DE REDE .....	11
5.2	APLICAÇÃO TOPOLOGIA DE REDE .....	11
5.2.1	TOPOLOGIA FISICA .....	11
5.2.1.1	<i>DISPOSIÇÃO FISICA</i> .....	13
5.3	CLASSES E DISTRIBUIÇÃO DA REDE .....	13
5.4	CONFIGURAÇÃO IP DA REDE .....	14
6	ÉTICA E LEGISLAÇÃO .....	16
6.1	O QUE É ÉTICA? .....	16
7	SOFTWARE .....	18
7.1	SOBRE O SOFTWARE .....	18
7.2	MANUAL DE USUARIO .....	20
7.3	LINHA DE CODIGO DO SISTEMA CTECH .....	51
8	CONCLUSÃO .....	99
	REFERÊNCIAS .....	100

## **1 INTRODUÇÃO**

Com as mudanças dos últimos tempos, o home office é uma opção cada vez mais viável e econômica para as empresas, tornando os funcionários menos dependente dos recursos da companhia.

A Startup Ctech foi criada para facilitar o dia a dia de pessoas e empresas que não possuem tempo livre, ou optaram por terceirizar o serviço de configurar, concertar e fazer manutenção de seus equipamentos, ganhando com isso tempo e diminuindo custos, garantindo com isso suporte, manutenção adequada e garantia.

Visando isso, e levando em consideração as urgências individuais, a Ctech se equipou, tanto em mão de obra qualificada, quanto em equipamentos modernos, para que possa entregar com qualidade um serviço resolvido em 24 horas.

Contando com o sucesso da ideia a empresa visa a expansão no mercado.

Neste projeto, apresentamos a empresa e um software no qual facilita e agiliza as atividades dentro da empresa.

### **1.1 O QUE É STARTUP?**

Não é fácil para as pessoas identificarem uma Startup, porém de acordo com inúmeros empreendedores e estudiosos, uma definição se sobressaiu como a mais completa. Ela foi feita por Eric Ries, considerado um dos maiores expoentes do assunto no mundo.

“Uma Startup é um grupo de pessoas à procura de um modelo de negócios repetível e escalável, trabalhando em condições de extrema incerteza.”

## **2 DESENVOLVIMENTO**

### **2.1 SOBRE A STARTUP**

A Ctech está em expansão e juntamente com esse crescimento existe a necessidade de se reformular e investir, com isso a empresa recebeu um aporte financeiro de um investidor anjo apostando no crescimento e na rentabilidade da empresa, sendo assim além de investir em equipamentos, também investiu em um sistema que faça todo gerenciamento das tarefas.

Esse investimento fez com que a empresa se estruturasse em um endereço no meio de centros comerciais, grandes empresas, consultórios médicos e afins, visando atingir um público maior e geral, não sendo apenas pessoas jurídicas, como também pessoas físicas e pequenos e médios estabelecimentos.

A empresa está localizada na Av. Ana Costa, Nº 222, 6º e 7º andar, funcionará de segunda à sexta-feira das 09:00 às 18:00, a mesma é composta por dois conjuntos em andares diferentes, cada conjunto composto por 3 salas, 2 banheiros e uma pequena copa.

No 6º andar localiza-se a parte operacional da empresa, composta por uma recepção, e duas salas operacionais, nesse andar há 5 funcionários, sendo 4 técnicos e 1 recepcionista, o mesmo está destinado ao atendimento ao público, entrega e retirada de serviços, bem como recepção e direcionamento de pessoas para reuniões com as gerencias e até mesmo contratação de possíveis mão de obra.

No 7º andar localiza-se a parte gerencial da empresa, composta por uma sala de reuniões e duas salas gerencias, nesse andar ficam apenas os gerentes.

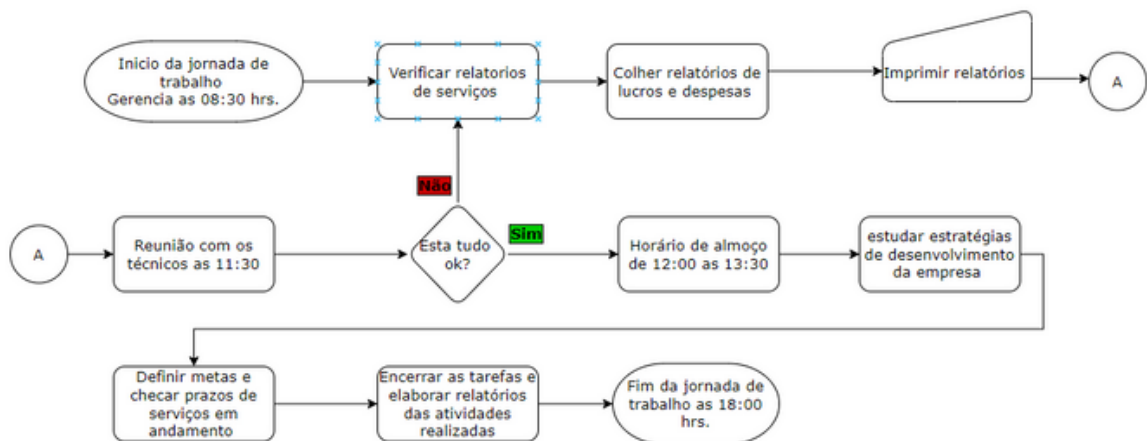
Por estarem em andares diferentes, o meio de comunicação utilizado é o Microsoft Teams, que permite a criação de um ambiente que unifica todos os colaboradores e time de gestão em um só lugar, além de possibilitar reuniões ao vivo, gravação e upload de vídeo, chats interativos, edição de documentos simultaneamente.

### 3 FUNCIONARIOS

#### 3.1 GERENTES

A empresa é composta por 2 gerentes, um comercial, que atua na parte de capacitação de cliente, negociações, estratégia e marketing, e o outro fica mais voltado para a parte financeira e administrativa, contratação de funcionários, avaliação de compras de equipamentos, fechamento de fluxo de caixa, investimentos e folha de funcionários.

Fluxograma 1 — gerentes



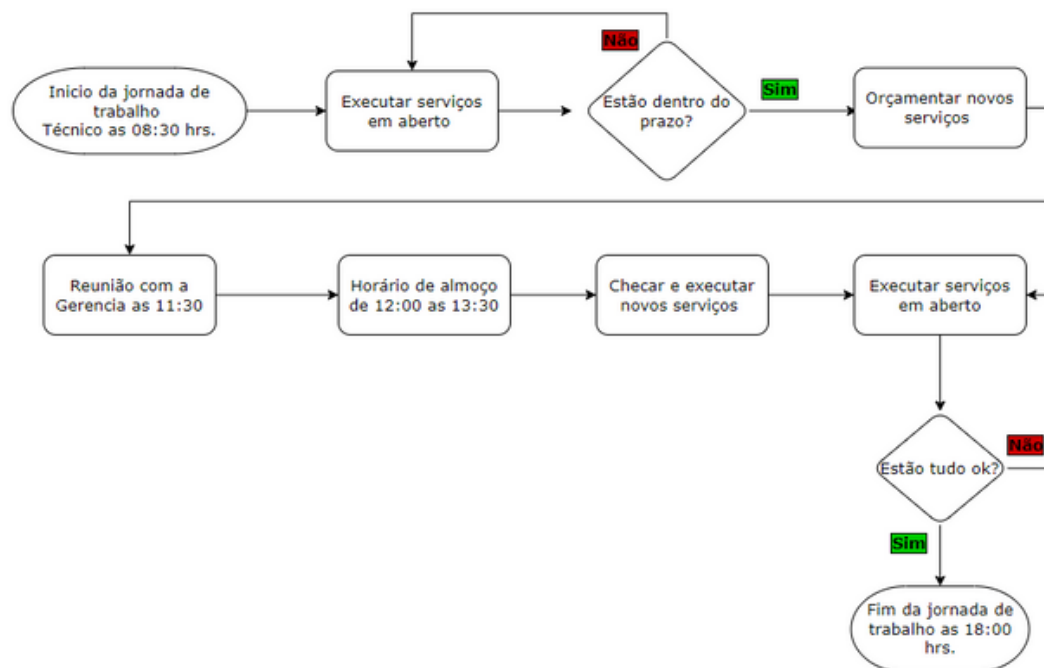
Fonte: Os autores (2022).

#### 3.2 TÉCNICOS

Os técnicos são responsáveis por avaliar os serviços, estipular prazos e valores assim como a execução dos serviços.



Fluxograma 2 — TÉCNICOS

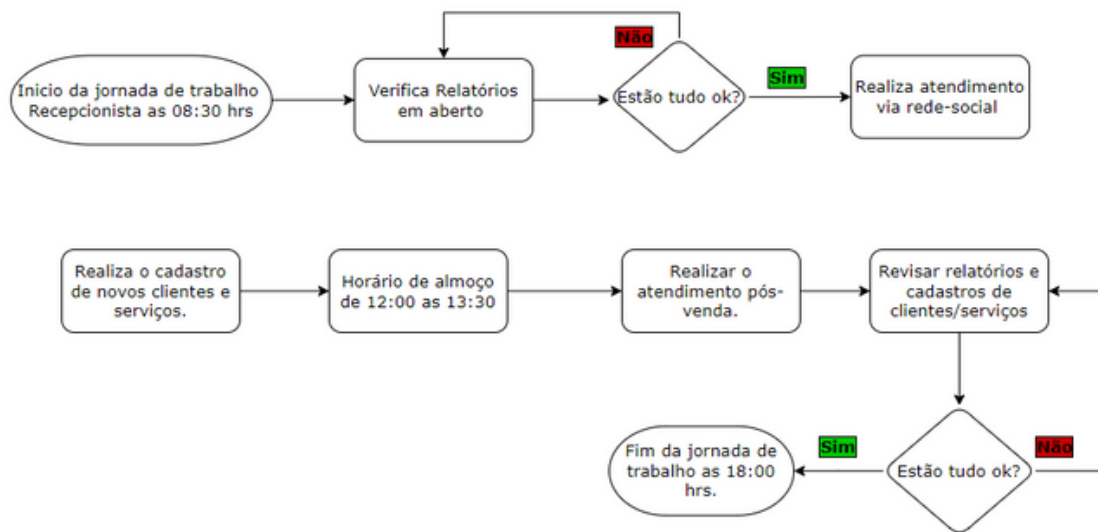


Fonte: Os autores (2022).

### 3.3 RECEPCIONISTA

Recepcionista é responsável pelo atendimento de modo geral, cadastrar clientes, conferir a ordem de serviços e receber os pagamentos.

Fluxograma 3 — RECEPCIONISTA



Fonte: Os autores (2022).

#### 4 PROSPECÇÃO FINANCEIRA

A Ctech, recebeu um aporte financeiro de R\$ 1 milhão de reais, se comprometendo que em dois anos, devolveria o investimento com 10% a mais por cada ano.

Atualmente é empresa tem 50 contratos fechados, com prazo mínimo de 48 meses, cada contrato mensal custa R\$ 3.500,00, garanto um total anual de contratos vigentes R\$ 2.100.000,00, visando um crescimento de 20% de contratos para o próximo semestre, a empresa almeja um ganho estimado líquido de R\$ 1.500.000,00 anual.

Tabela 1 — valores estimados

MESES	FUNCIONARIOS (R\$ 7.000,00x5)	ALUGUEL	CONTAS BASICAS	GASTOS MEDIOS MENSAIS
JANEIRO	R\$ 35.000,00	R\$ 10.000,00	R\$ 5.000,00	R\$ 50.000,00
FEVEREIRO	R\$ 35.000,00	R\$ 10.000,00	R\$ 5.000,00	R\$ 50.000,00
MARÇO	R\$ 35.000,00	R\$ 10.000,00	R\$ 5.000,00	R\$ 50.000,00
ABRIL	R\$ 35.000,00	R\$ 10.000,00	R\$ 5.000,00	R\$ 50.000,00
MAIO	R\$ 35.000,00	R\$ 10.000,00	R\$ 5.000,00	R\$ 50.000,00
JUNHO	R\$ 35.000,00	R\$ 10.000,00	R\$ 5.000,00	R\$ 50.000,00
JULHO	R\$ 35.000,00	R\$ 10.000,00	R\$ 5.000,00	R\$ 50.000,00
AGOSTO	R\$ 35.000,00	R\$ 10.000,00	R\$ 5.000,00	R\$ 50.000,00
SETEMBRO	R\$ 35.000,00	R\$ 10.000,00	R\$ 5.000,00	R\$ 50.000,00
OUTUBRO	R\$ 35.000,00	R\$ 10.000,00	R\$ 5.000,00	R\$ 50.000,00
NOVEMBRO	R\$ 35.000,00	R\$ 10.000,00	R\$ 5.000,00	R\$ 50.000,00
DEZEMBRO	R\$ 35.000,00	R\$ 10.000,00	R\$ 5.000,00	R\$ 50.000,00
<b>TOTAL</b>	<b>R\$ 420.000,00</b>	<b>R\$ 120.000,00</b>	<b>R\$ 60.000,00</b>	<b>R\$ 600.000,00</b>

Fonte: Os autores (2022).

## **5 REDES**

### **5.1 CONCEITO DE TOPOLOGIA DE REDE**

A topologia rede é o termo usado para definir a forma como se estrutura a rede de computadores. Ela determina a disposição das máquinas entre si, e os elementos que se conectam a ela. Toda essa organização interfere diretamente no comportamento, estabilidade e qualidade da rede. Ela se divide em dois ramos: topologia física e topologia lógica. Onde a topologia física é responsável pela conexão física da rede e a lógica corresponde a maneira como os dados irão trafegar pela rede, com isso, abaixo descrevemos como ficou a topologia logica e física da Ctech.

### **5.2 APLICAÇÃO TOPOLOGIA DE REDE**

Como ponto inicial é preciso conectar dois andares, o desafio é comunicar cada local com uma conexão segura, estável e ajustar a rede.

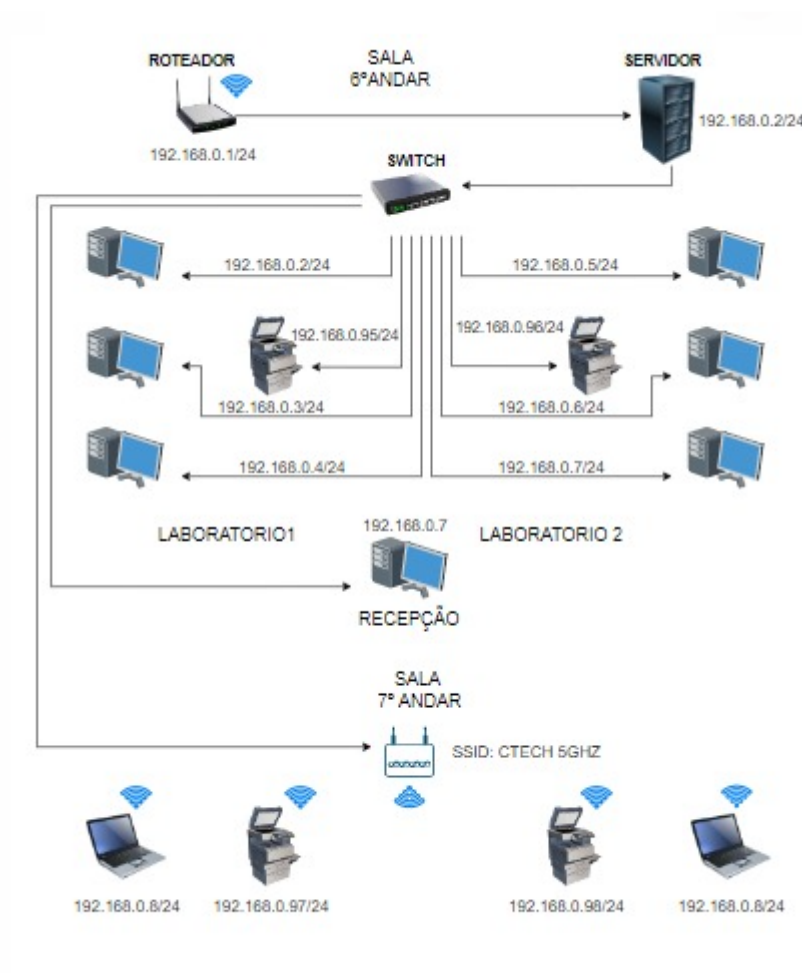
A utilização de um roteador com uma entrada fast Ethernet padrão e outra fast Ethernet de fibra óptica tem a ver com a segurança da rede e dos dados enviados e recebidos, que no caso dos cabos de cobre, por exemplo, é mais fácil que um hacker consiga interceptar algum tipo de informação, já que a transmissão emite um tipo de sinal rastreável, o que facilita o acesso não autorizado à rede, ja a fibra óptica com sua qualidade, se torna de fácil visualização quando existe qualquer tipo de invasão à rede, isso porque quando a luz acaba vazando da fibra óptica é gerado um dano a todo o sistema, o que facilita a percepção de que está ocorrendo algum problema relacionado ao possível vazamento de informações, sendo assim a conexão dedicada deverá ser através do mesmo para se obter uma velocidade na casa do gigabit através do provedor, ao qual será utilizada para conectar o roteadores e a primeira ponta da rede estrela.

#### **5.2.1 TOPOLOGIA FISICA**

A melhor topologia para ambos os locais é a estrela e por esse motivo é a escolhida para o projeto, não apenas por ser a que melhor se encaixa, como também ser o padrão que o mercado vem adotando com o tempo, essa topologia suporta falhas na rede e é de fácil instalação e possibilidade de expansão, com

melhor gerenciamento e monitoramento.

Fluxograma 4 — TOPOLOGIA FISICA



Fonte: Os autores (2022).

Um dos pontos negativos que é muito relevante nesse tipo de rede é que por se tratar de uma rede centralizada caso o switch venha a apresentar defeito a rede toda é comprometida, tornando a substituição da peça algo de urgência.

Já como vantagens temos o desempenho, escalabilidade e alta disponibilidade mais seguros, ao trabalharmos com etapas de consultar e manipular ou armazenar as informações conclusões dos processos lógicos.

Essa visão física nos dá o resultado de conexão a todas as máquinas e equipamentos disponíveis, utilizando estratégia de organização, cabeamento, e disposição das máquinas.

Como protocolo de rede o projeto utilizará o TCP/IP que permite padronizar o projeto o que leva a uma maior compatibilidade de dados entre todos os dispositivos.

Esse protocolo tem a grande vantagem de estar disponível na maioria dos sistemas operacionais.

#### 5.2.1.1 *DISPOSIÇÃO FÍSICA*

O cenário disponível propõe os seguintes recursos que serão listados a seguir, assim, demarcando as disposições dos locais dessa infraestrutura.

No 6º andar ficará:

- 01 Roteador que será o Gateway da rede, que fara conexão com o Switch, usará fibra óptica com conectores SFP +;
- 01 Switch de 24 portas será utilizado para conectar todos os demais equipamentos, o mesmo usará cabeamento de STP (Par Trançado com Blindagem categoria 6, que já conta com uma velocidade de 1000 Mbps que atende melhor todas as portas que irão buscar dados direto no servidor);
- 01 (um) servidor responsável por manter os serviços: DNS, arquivos dos usuários, serviço de diretórios (Microsoft Active Directory: AD), servidor de impressão, antivírus: Kaspersky (endpoint), rotinas e pesquisas através da internet, arquivos dos usuários e servidor de impressão;
- 03 (tres) impressoras multifuncionais em rede;
- 05 (cinco) estações de trabalho (hosts) distribuídas entre desktops.

Assim ficando com os restantes de suas portas disponíveis para futuros servidores ou impressoras.

No 7º andar, a estrutura fica um pouco mais enxuta, será passado um cabeamento de STP categoria 6, via conduíte da estrutura do prédio, para conectar:

- 1 (um) Access Point (AP), que conectará os demais equipamentos;
- 2 (duas) impressoras multifuncionais em rede;
- 2 (duas) estações de trabalho (hosts) distribuídas entre notebooks;

### 5.3 **CLASSES E DISTRIBUIÇÃO DA REDE**

Um endereço de IPv4 é formado por 32bits, e dividido em quatro octetos representado na forma de decimal (192.168.0.1). Uma parte representa o endereço da rede e o outra o dispositivo dentro da rede. Com isso é possível classificar em cinco classes diferentes, sendo elas as Classes A, B, C, D e E.

A Classe C que deve ser utilizada para o projeto e gerido pela rede é a classe C onde ela pode contar com 254(28-2) de endereços por dispositivo na rede. Como nenhuma das duas localidades exige mais que isso de endereço, a Classe C corresponde mais que suficiente para implementação do projeto.

O número de rede escolhido para a central é o 192.168.0.0 e a distribuição do endereço lógico ficará a cargo do servidor de DNS, seu endereço de IP será o 192.168.0.2. O serviço de DHCP, ficará configurado da seguinte maneira, Gateway padrão 192.168.0.1, servidor de DNS que se encontra nesse próprio servidor 192.168.0.2 e a distribuição de endereços irá iniciar a partir do ip address 192.168.0.10, assim deixando 8 endereços disponíveis para atribuição manual de endereços a rede caso novos projetos que venha a necessitar.

O número máximo de usuários foi configurado com base na quantidade de dispositivos elevado a 2. Com isso as 5 impressoras são adicionadas manualmente no endereço. O servidor ficará responsável por distribuir os endereços das estações de trabalho. A conexão entre as duas redes será pela rede 192.168.20.0/24.

#### 5.4 CONFIGURAÇÃO IP DA REDE

Os endereços seguirão a seguinte configuração, estações de trabalho, serão configuradas automaticamente pelo DHCP tendo como início o endereço 192.168.0.10, e o número máximo de usuário em 100.

Na Tabela apresenta-se a lista de como irão ficar os endereçamentos dos IPs Address.

Tabela 2 — Tabela - Endereços IPs (continua)

Dispositivos	IP Address
Roteador - Gateway	192.168.0.1/24
Servidor DNS	192.168.0.2/24
Impressora 1	192.168.0.95/24
Impressora 2	192.168.0.96/24
Impressora 3	192.168.0.97/24
Impressora 4	192.168.0.98/24
Impressora 5	

Tabela 2 — Tabela - Endereços IPs (conclusão)

Dispositivos	IP Address
	192.168.0.99/24
Desktop 1	192.168.0.3/24
Desktop 2	192.168.0.4/24
Desktop 3	192.168.0.5/24
Desktop 4	192.168.0.6/24
Desktop 5	192.168.0.7/24
Notebook 1	192.168.0.8/24
Notebook 2	192.168.0.9/24

Fonte: Os autores (2022).



## 6 ÉTICA E LEGISLAÇÃO

### 6.1 O QUE É ÉTICA?

São os princípios que você e eu usamos para responder ao “Quero? Devo? Posso?”.

A Ética é o que marca a fronteira da nossa convivência. Seja com as outras pessoas ou com o mercado, Ética é aquela perspectiva para olharmos os nossos princípios e os nossos valores para existirmos juntos.

Em nosso mundo mega conectado, um número enorme de dados está se misturando ao reconhecimento de padrões, aprendizado de máquinas, algoritmos e outros softwares inteligentes para dar suporte a um novo nível de computação, com isso existe uma relação de confiança entre o consumidor e as empresas com as quais ele interage.

Os consumidores fornecem seus dados e esperam que suas informações pessoais estejam protegidas e sejam utilizadas de forma adequada, para manter a confiança dos clientes, as organizações devem passar a implementar uma estrutura de ética digital. O ideal é que ela seja desenvolvida para definir não apenas como uma empresa inova e negocia com seus clientes, mas também como as informações sobre funcionários são utilizadas e gerenciadas.

Para a maioria das organizações, o primeiro passo deste processo é aumentar a conscientização sobre o que é ética digital, e ensinar as partes envolvidas e os funcionários que isso não é a mesma coisa que compliance, privacidade ou segurança.

Entretanto ações realizadas quando no exercício de sua profissão, isso não o exime das responsabilidades legais, o direito é um conjunto de normas que devem ser seguidas pelos cidadãos, independentemente de existir ou não regulamentação do exercício profissional em informática, o que devemos entender é que a ética é imprescindível para a vida e o respeito a ela deve ser evidente no exercício de qualquer profissão.

Em suma, LGPD significa Lei Geral de Proteção de Dados (13.709),

regulariza o tratamento de dados pessoais sensíveis, em meios off e digitais. Por pessoa natural ou por pessoa jurídica de direito público ou privado. Seu objetivo é proteger os direitos fundamentais de liberdade e de privacidade e o livre desenvolvimento da personalidade da pessoa natural.

Na prática, a LGPD define a obrigação das empresas perante o tratamento de dados pessoais. Além de entender e adequar onde estão os dados pessoais dos clientes, colaboradores e terceiros, é necessário demonstrar os mecanismos de proteção existentes e as políticas aplicadas e divulgadas dentro da empresa.

A incapacidade de avaliar adequadamente as considerações da ética de dados, colocará a reputação das empresas em risco considerável na economia. Da mesma forma, em nossa sociedade cada vez mais conectada e com quantidade de dados cada vez maior, o uso e gerenciamento ético de dados de funcionários será uma prioridade para as empresas que busquem atrair e reter os melhores talentos.

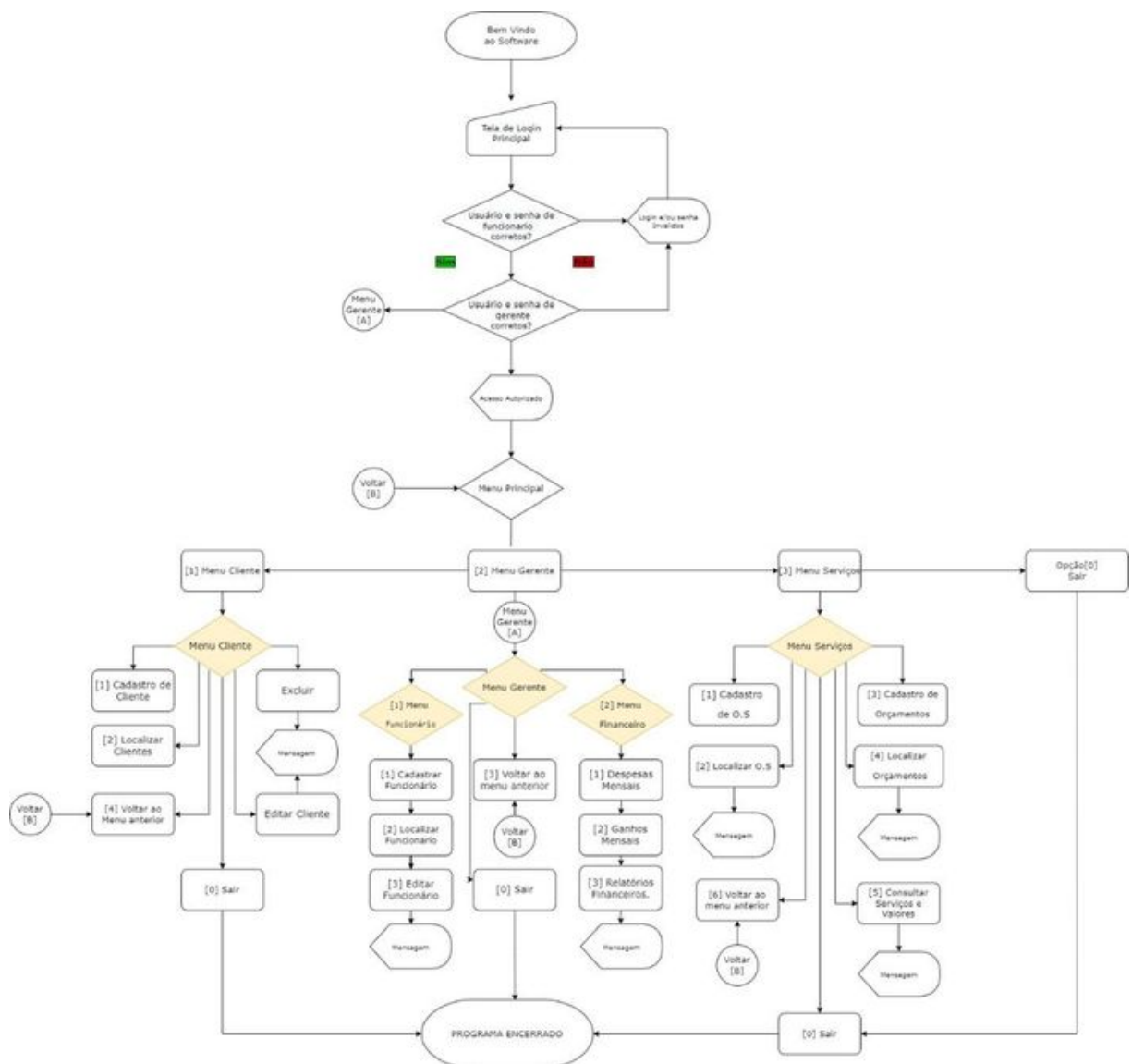
## **7 SOFTWARE**

### **7.1 SOBRE O SOFTWARE**

A necessidade inicial da empresa é um software que gerencie as tarefas da Startup, emissão de orçamentos, ordens de serviços, cadastros de receitas e despesas, cadastros de cliente e relatórios gerenciais, levando em consideração o novo cenário em que a empresa está se inserindo para que consigam atender seu plano de negócios.

No decorrer do trabalho, informamos a como será utilizado o software por cada colaborador, abaixo fluxograma do funcionamento do software e manual do usuário

Fluxograma 5 — FLUXOGRAMA DO SOFTWARE



Fonte: Os autores (2022).

## 7.2 MANUAL DE USUARIO

Figura 1 — TELA DE LOGIN PRINCIPAL



Fonte: Os autores (2022).

A mesma é responsável por realizar o controle de acesso para o uso do sistema interno da empresa, tendo 2 tipos de acesso, acesso a recepcionista/técnico direcionando para o menu principal e outra de gerência que direcionando direto para o menu gerente.

Imagem 1 — TELA DE LOGIN PRINCIPAL COM ACESSO AUTORIZADO



Fonte: Os autores (2022).

Nessa tela, ao realizar o acesso com usuário e senha correto, irá aparecer a mensagem informando que foi logado como funcionário e após 1 segundo será redirecionada a tela de menu principal ou será informado que foi logado como gerente e após 1 segundo será redirecionada ao menu gerente de acordo com login que fará o acesso.

Figura 2 — TELA DE LOGIN PRINCIPAL COM ACESSO NÃO AUTORIZADO



Fonte: Os autores (2022).

Nessa tela, ao realizar o acesso com usuário e senha incorreto, irá aparecer a mensagem informando que foi o usuário e/ou senha estão inválidos e será redirecionada a tela de login novamente.

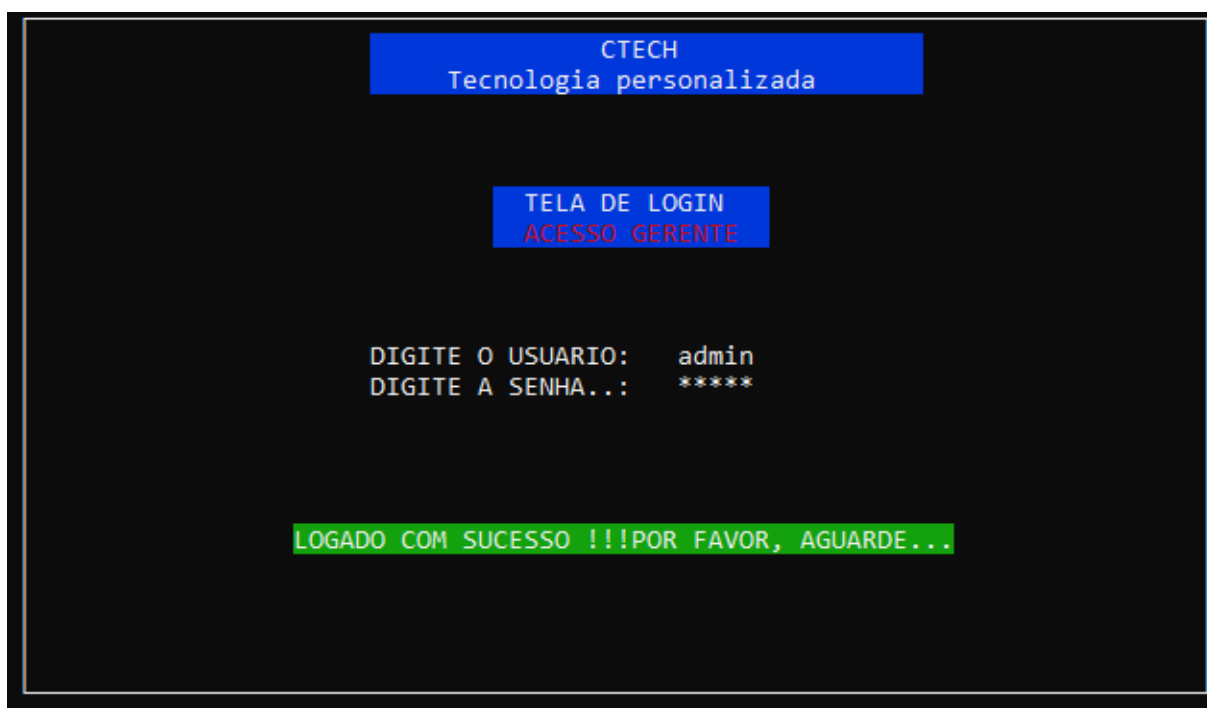
Imagem 2 — TELA DE LOGIN GERENTE



Fonte: Os autores (2022).

A mesma é responsável por realizar o controle de acesso exclusivo a gerência para o uso do sistema interno da empresa.

Imagem 3 — TELA DE LOGIN DO GERENTE COM ACESSO AUTORIZADO



Fonte: Os autores (2022).



Nessa tela, ao realizar o acesso com usuário e senha correto, irá aparecer a mensagem informando que foi logado com sucesso e após 1 segundo será redirecionada a tela de menu gerente.

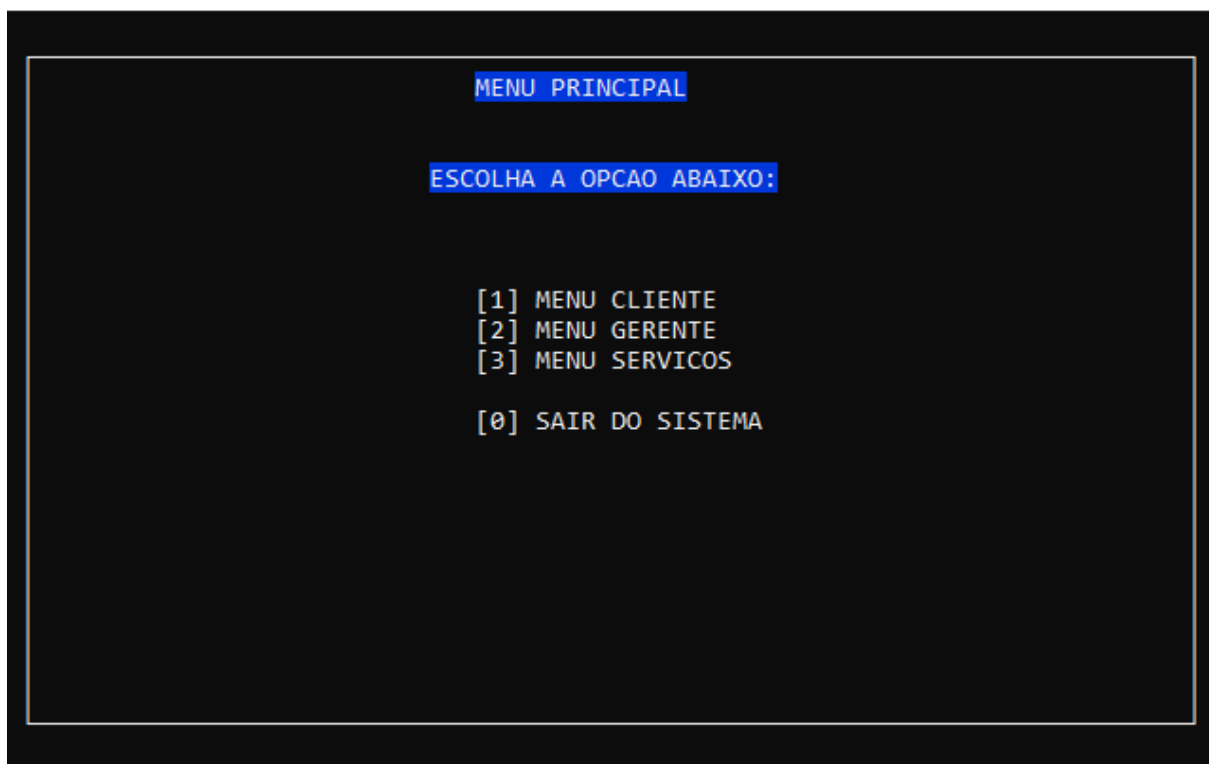
Imagem 4 — TELA DE LOGIN DO GERENTE COM ACESSO NEGADO



Fonte: Os autores (2022).

Nessa tela, ao realizar o acesso com usuário e senha do gerente incorreto, aparecerá a mensagem de usuário ou senha estão incorretos e será redirecionada ao menu principal.

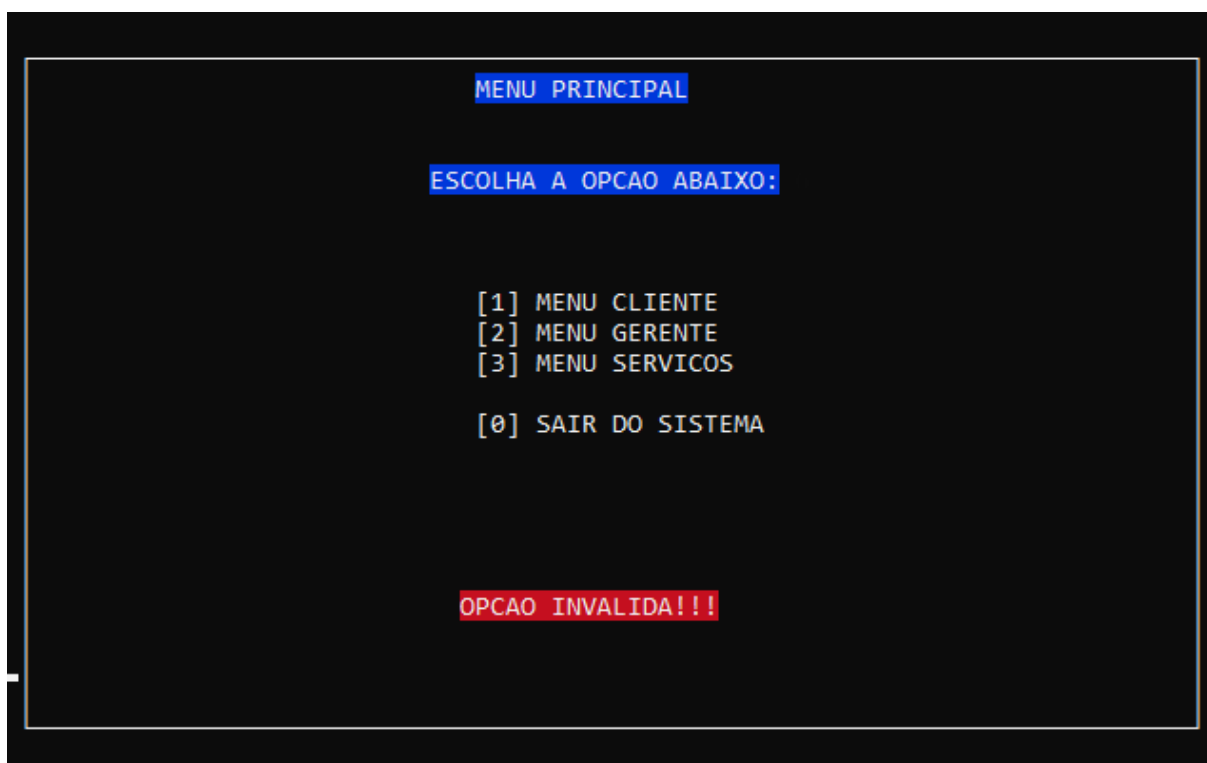
Imagem 5 — TELA DE MENU PRINCIPAL



Fonte: Os autores (2022).

A mesma é responsável por fazer o direcionamento para a opção desejada.

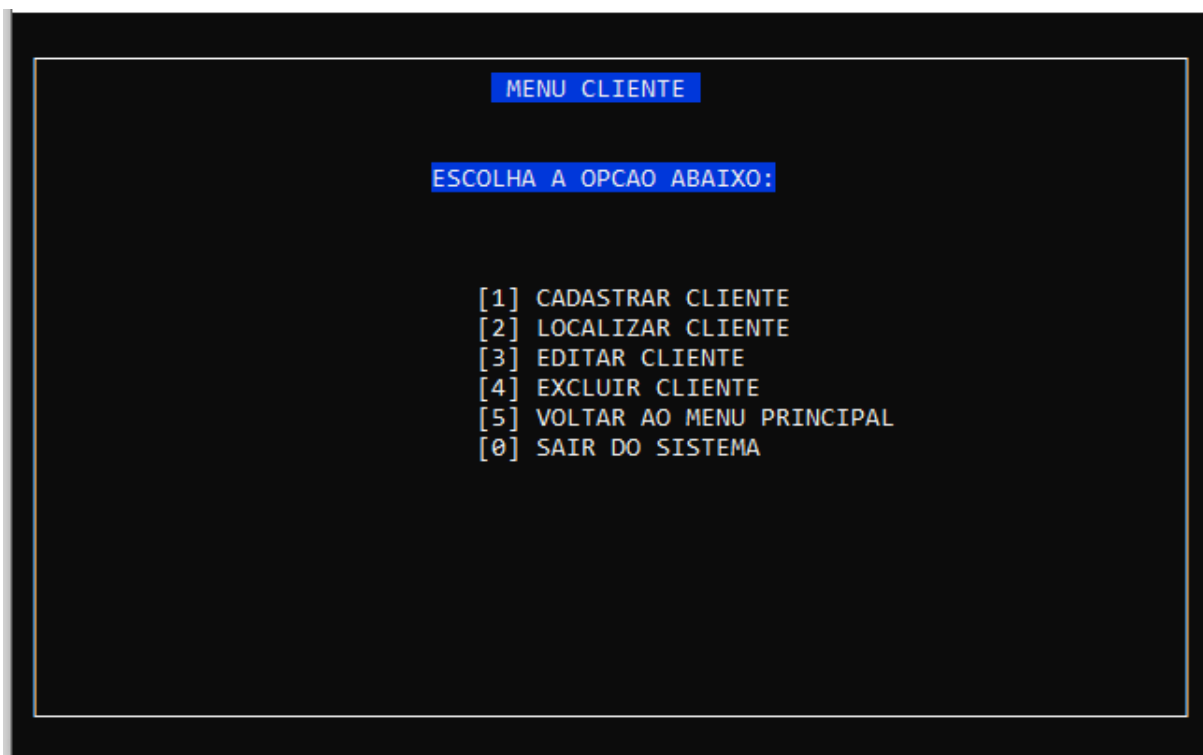
Imagem 6 — TELA DE ERRO AO DIGITAR A OPÇÃO NÃO CORRESPONDENTE



Fonte: Os autores (2022).

Nessa tela, ao apertar a opção inexistente, irá aparecer uma mensagem de 1 segundo em vermelho informando que a opção é inválida.

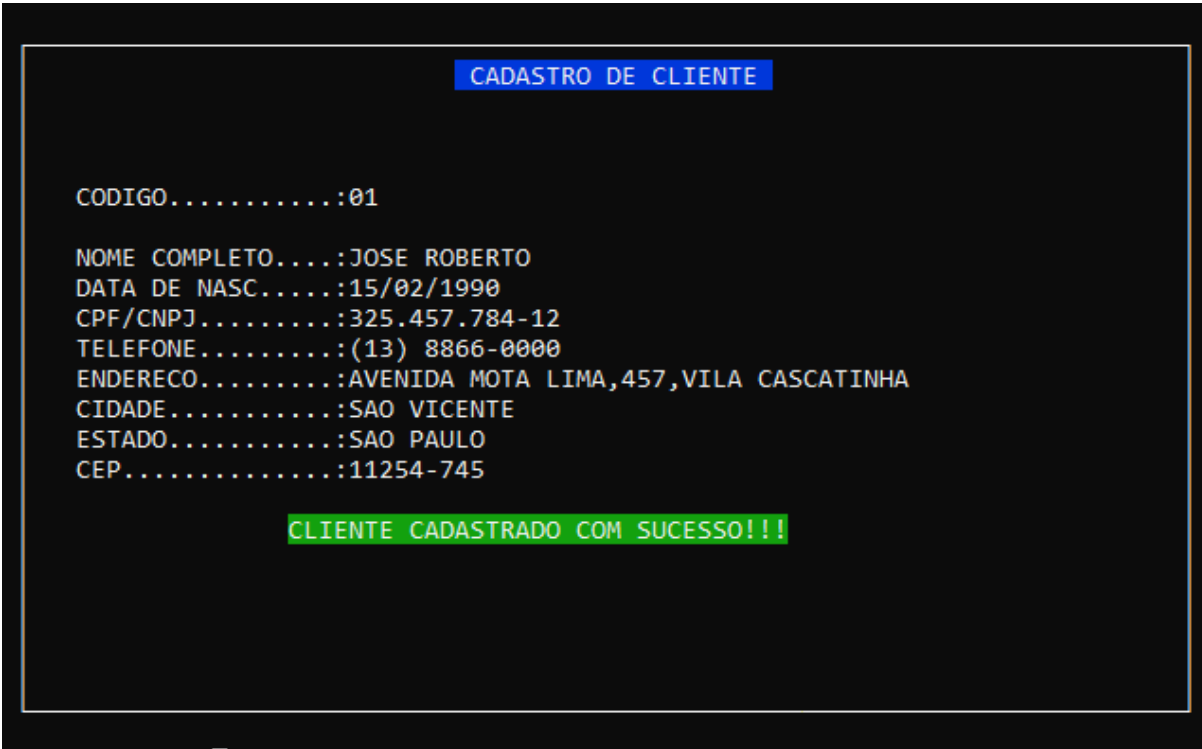
Imagem 7 — TELA MENU CLIENTE



Fonte: Os autores (2022).

Nessa tela, a recepcionista terá as opções de [1] cadastrar cliente, [2] localizar cliente, [3] editar cliente, [4] excluir cliente, [5] voltar ao menu principal e [0] sair do sistema.

Figura 3 — TELA DE CADASTRO DE CLIENTE



**CADASTRO DE CLIENTE**

CODIGO.....:01

NOME COMPLETO.....:JOSE ROBERTO

DATA DE NASC.....:15/02/1990

CPF/CNPJ.....:325.457.784-12

TELEFONE.....:(13) 8866-0000

ENDereco.....:AVENIDA MOTA LIMA,457,VILA CASCATINHA

CIDADE.....:SAO VICENTE

ESTADO.....:SAO PAULO

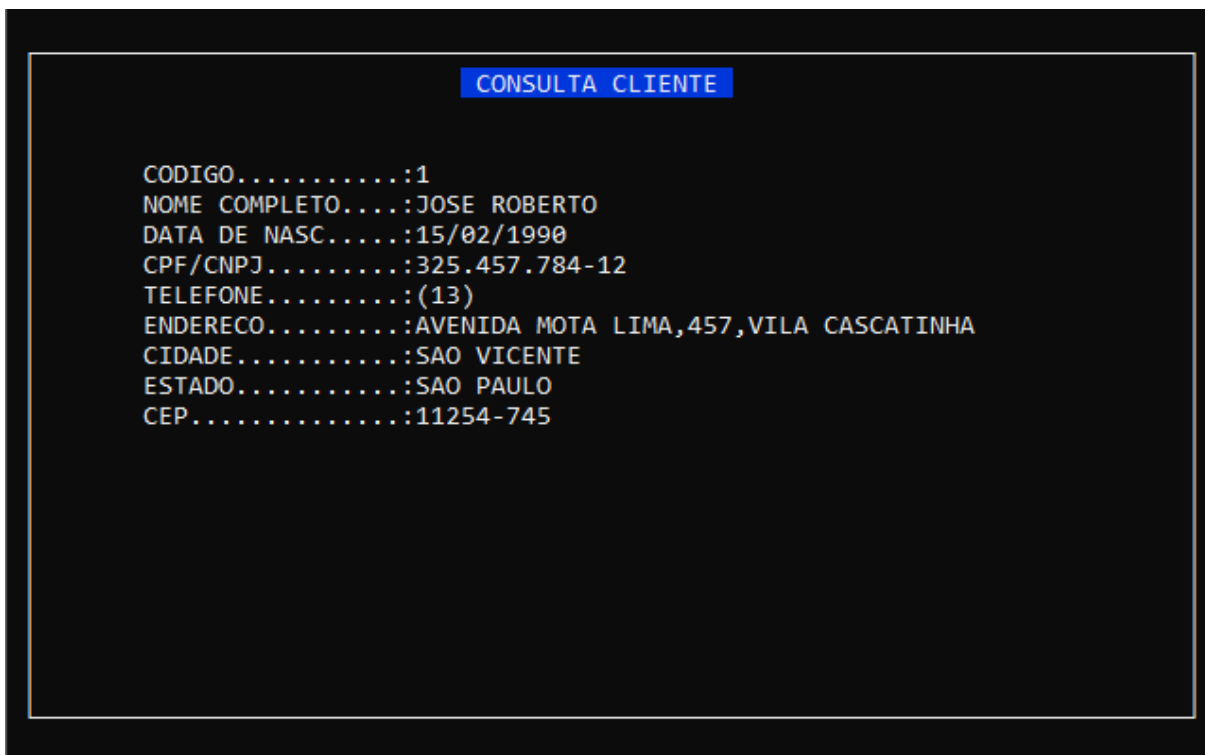
CEP.....:11254-745

**CLIENTE CADASTRADO COM SUCESSO!!!**

Fonte: Os autores (2022).

Na opção [1] cadastrar Cliente, visa a recepcionista obter os dados dos clientes afins de poder cadastrar a ordem de serviço, ter os dados para entrar em contato após a manutenção e gerar nota fiscal dos serviços realizados, sendo salvo em um arquivo com extensão "*cliente.txt*", após cadastrado a tela é redirecionada ao menu principal.

Figura 4 — TELA CONSULTA CLIENTE



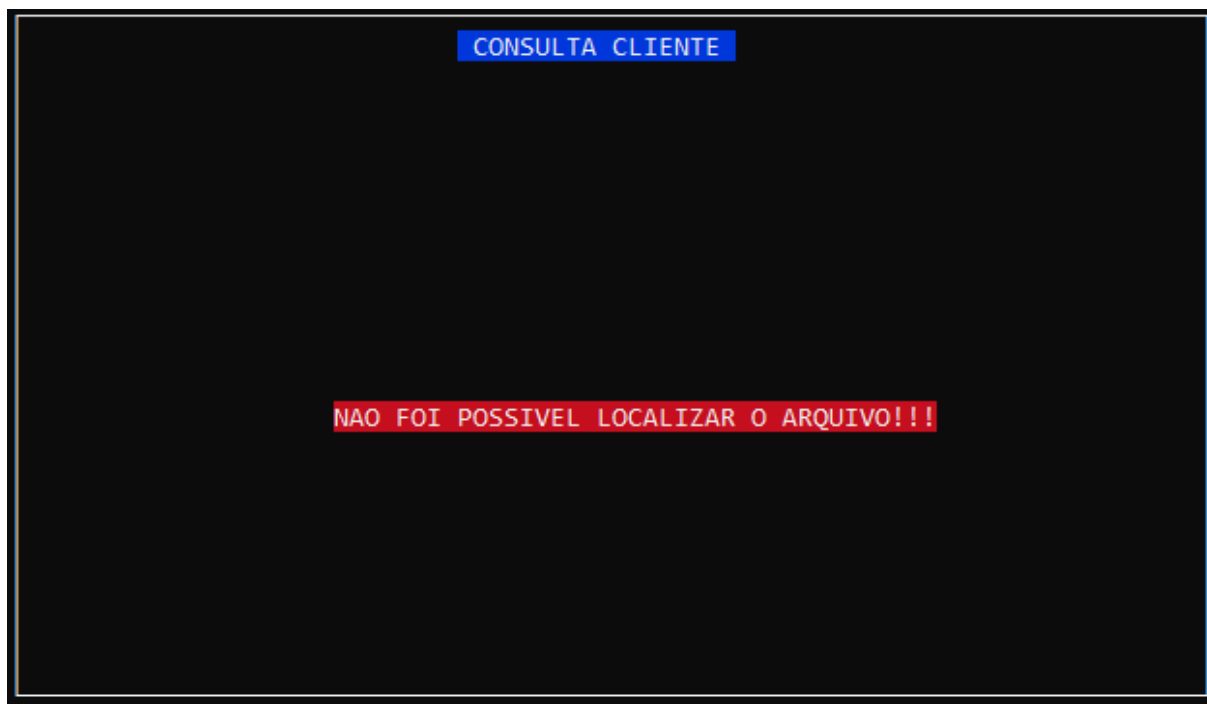
A screenshot of a terminal window with a black background and white text. At the top center, the title 'CONSULTA CLIENTE' is displayed in a blue rectangular box. Below the title, the following client information is listed in a plain text format:

```
CODIGO.....:1
NOME COMPLETO....:JOSE ROBERTO
DATA DE NASC.....:15/02/1990
CPF/CNPJ.....:325.457.784-12
TELEFONE.....:(13)
ENDereco.....:AVENIDA MOTA LIMA,457,VILA CASCATINHA
CIDADE.....:SAO VICENTE
ESTADO.....:SAO PAULO
CEP.....:11254-745
```

Fonte: Os autores (2022).

Na opção de localizar cliente, irá aparecer em lista os clientes cadastrados no arquivo "cliente.txt". Em breve terá atualizações, com a finalidade de melhoria do software, sendo aplicado consultar por CPF ou Nome do cliente.

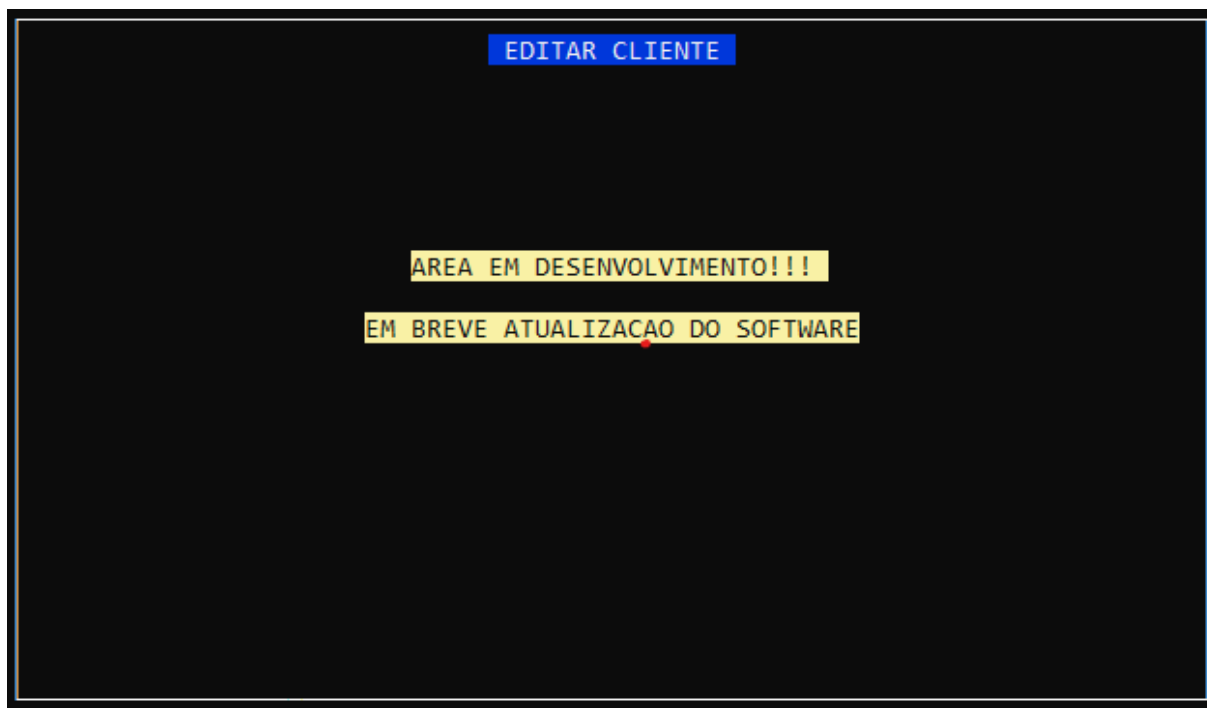
Figura 5 — TELA LOCALIZAR CLIENTE SEM TER O ARQUIVO



Fonte: Os autores (2022).

Nessa tela, ao tentar localizar o cliente e não existir o arquivo “*cliente.txt*”, será apresentado uma mensagem de erro em vermelho e em seguida voltará ao menu serviços.

Figura 6 — TELA EDITAR CLIENTE

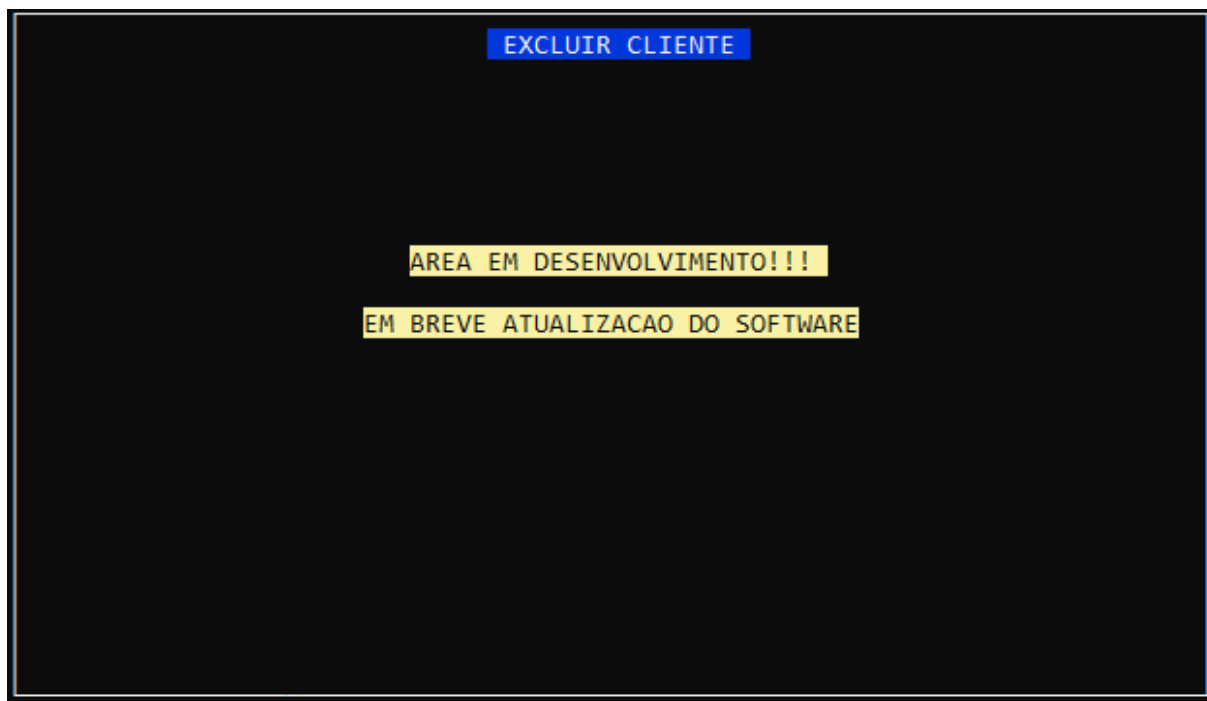


Fonte: Os autores (2022).

A opção de editar cliente visa a recepcionista realizar a atualização dos dados do cliente, porém, em breve terá uma atualização afins de melhoria do software, sendo aplicado a consulta por CPF ou nome, e em seguida aparecerá as opções a qual deseja alterar.



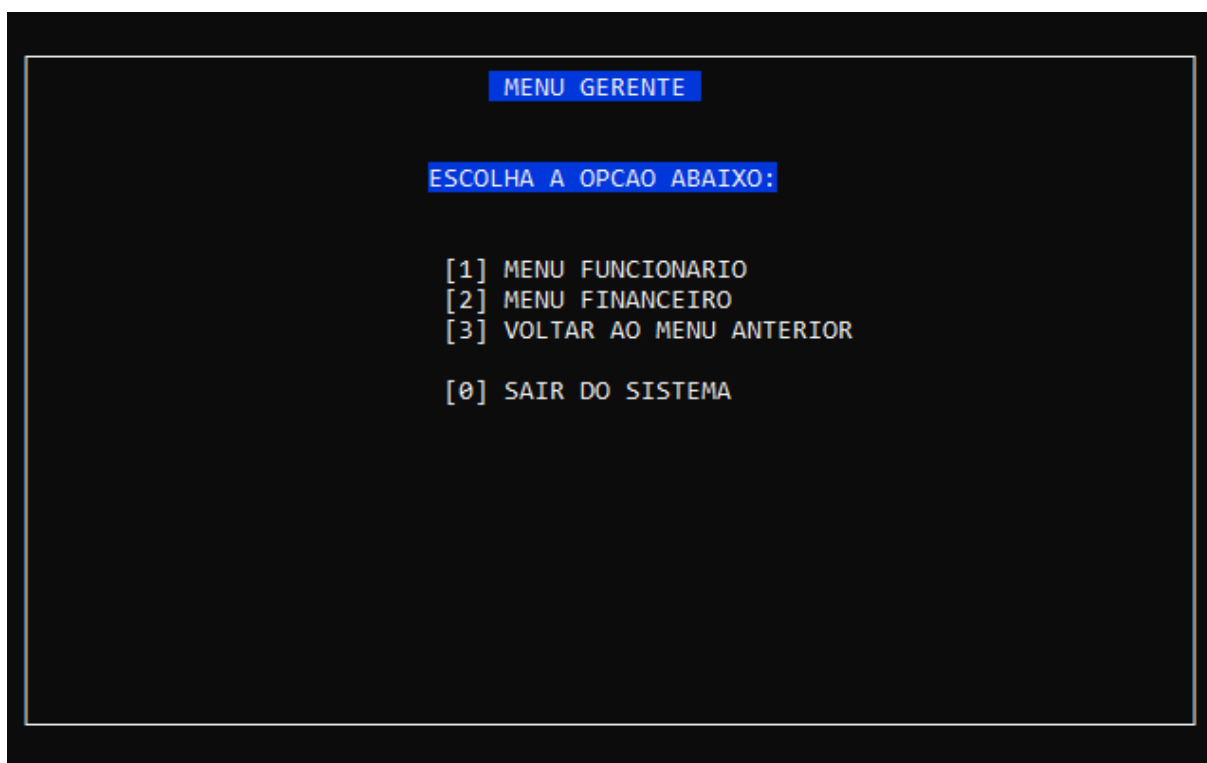
Figura 7 — TELA EXCLUIR CLIENTE



Fonte: Os autores (2022).

A opção de excluir cliente, visa localizar o mesmo por CPF ou Nome, e terá a finalidade de exclusão do cliente selecionado, porém, quem deverá validar a exclusão será com acesso do gerente.

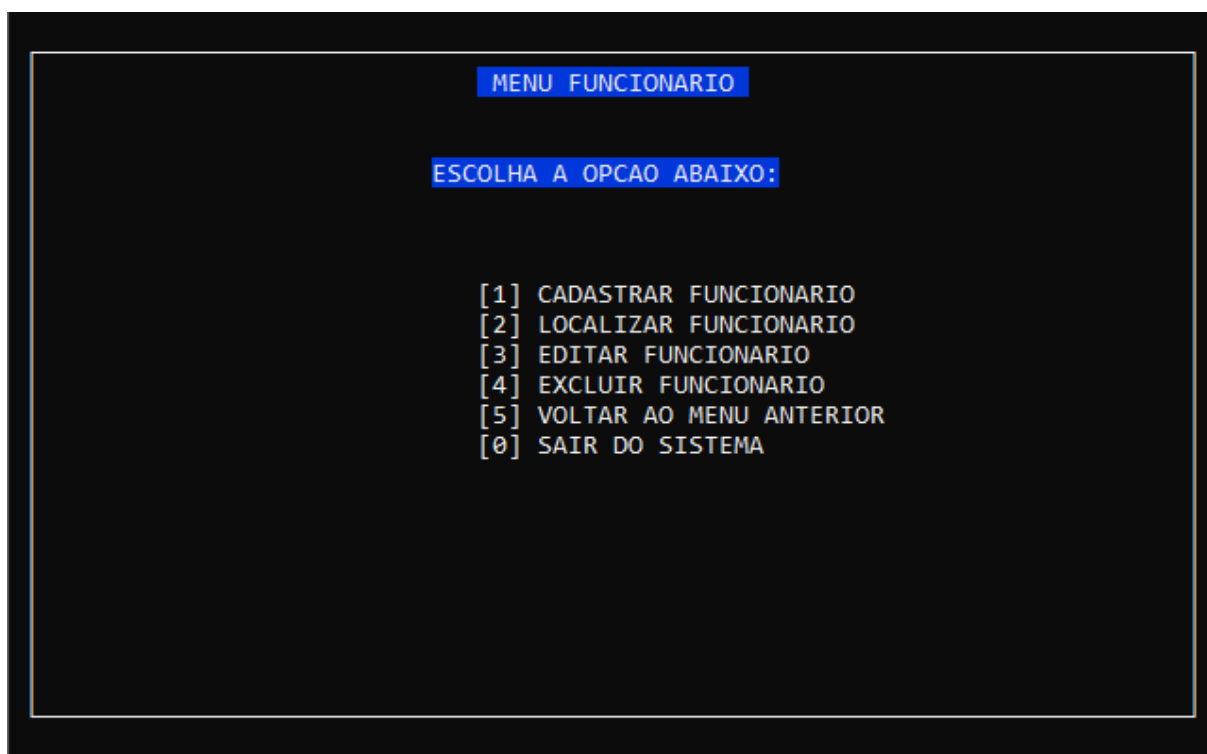
Figura 8 — TELA MENU GERENTE



Fonte: Os autores (2022).

Nessa tela, o gerente terá as opções de [1] menu funcionário, [2] menu financeiro, [3] voltar ao menu anterior e [0] sair do sistema.

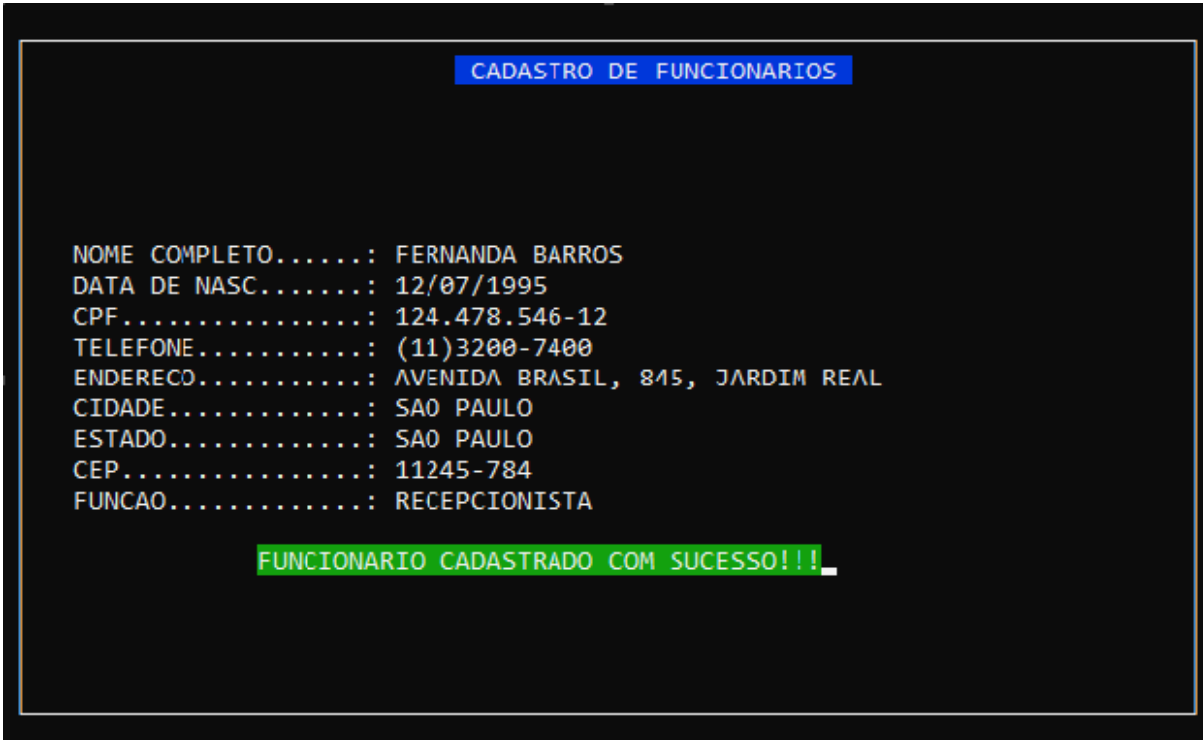
Figura 9 — TELA MENU FUNCIONÁRIOS



Fonte: Os autores (2022).

Nessa tela, o gerente terá as opções de [1] cadastrar funcionário, [2] localizar funcionário, [3] editar funcionário, [4] excluir funcionário, [5] voltar ao menu anterior e [0] sair do sistema.

Figura 10 — TELA CADASTRO DE FUNCIONÁRIOS



```
CADASTRO DE FUNCIONARIOS

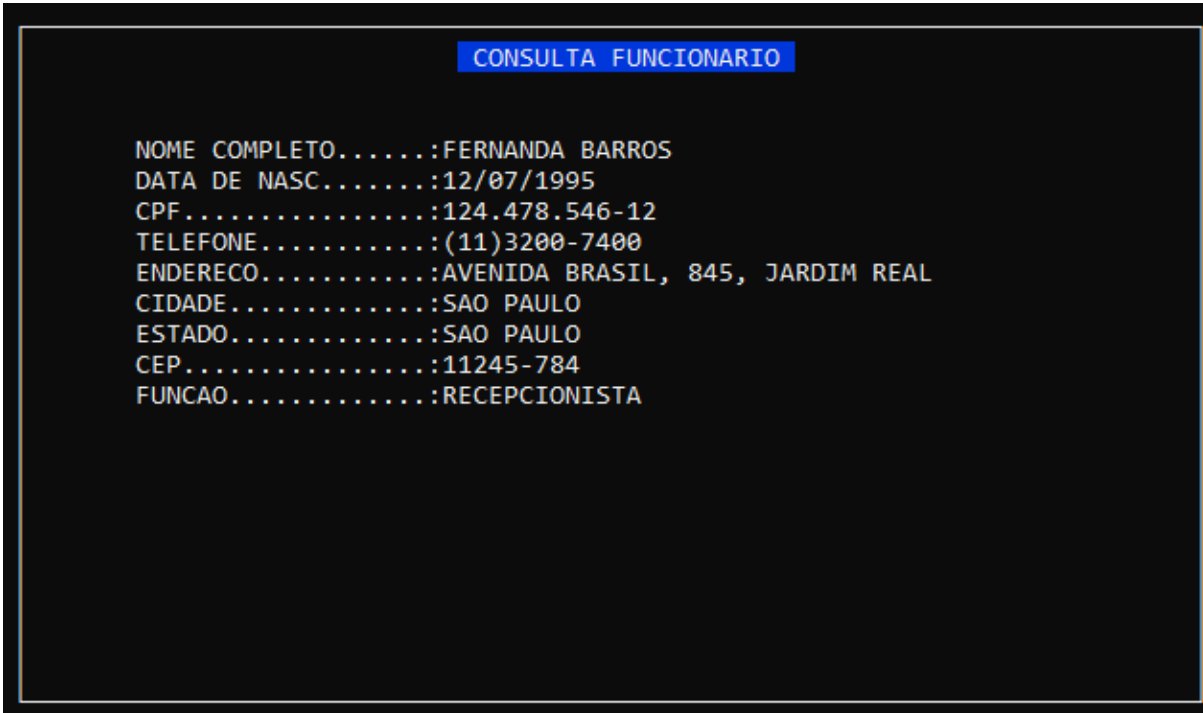
NOME COMPLETO.....: FERNANDA BARROS
DATA DE NASC.....: 12/07/1995
CPF.....: 124.478.546-12
TELEFONE.....: (11)3200-7400
ENDereco.....: AVENIDA BRASIL, 845, JARDIM REAL
CIDADE.....: SAO PAULO
ESTADO.....: SAO PAULO
CEP.....: 11245-784
FUNCAO.....: RECEPCIONISTA

FUNCIONARIO CADASTRADO COM SUCESSO!!!
```

Fonte: Os autores (2022).

A opção de [1] cadastrar funcionário visa o gerente obter os dados dos funcionários, sendo salvo em um arquivo com extensão "*funcionario.txt*", após cadastrado a tela é redirecionada ao menu gerente.

Figura 11 — TELA LOCALIZAR FUNCIONÁRIO



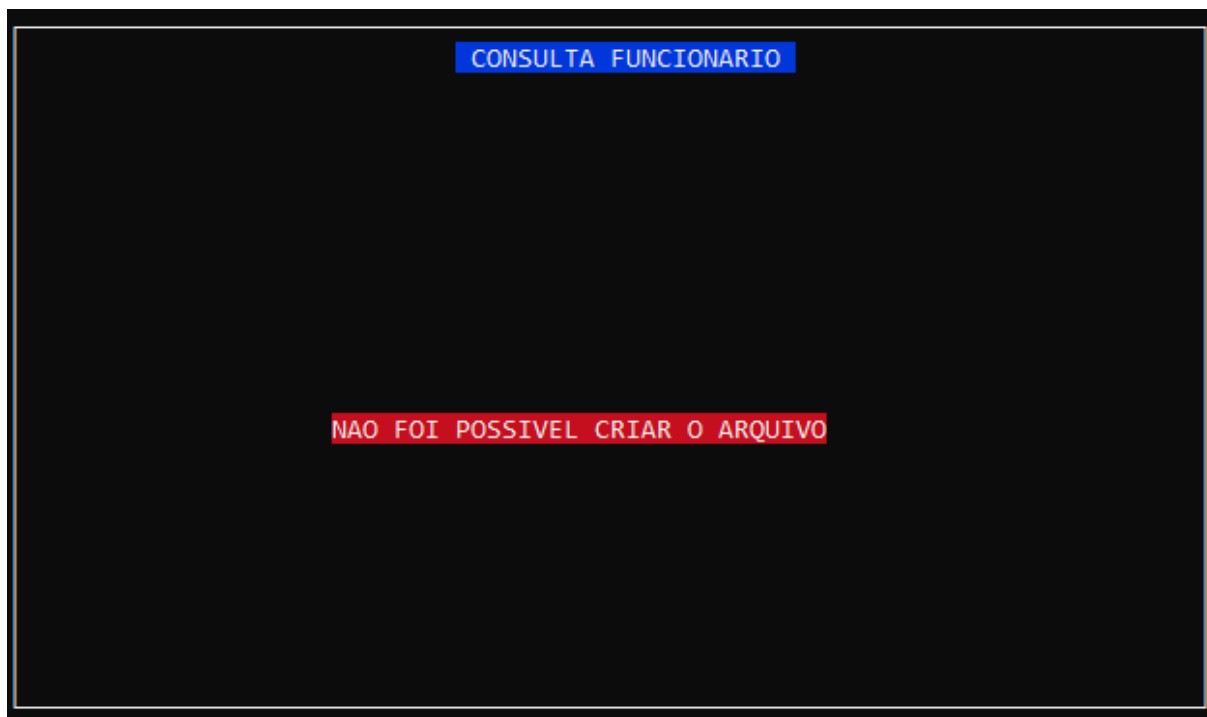
A screenshot of a terminal window with a black background and white text. At the top center, the title 'CONSULTA FUNCIONARIO' is displayed in a blue rectangular box. Below the title, the following employee information is listed in a plain text format:

NOME COMPLETO.....	FERNANDA BARROS
DATA DE NASC.....	12/07/1995
CPF.....	124.478.546-12
TELEFONE.....	(11)3200-7400
ENDERECO.....	AVENIDA BRASIL, 845, JARDIM REAL
CIDADE.....	SAO PAULO
ESTADO.....	SAO PAULO
CEP.....	11245-784
FUNCAO.....	RECEPCIONISTA

Fonte: Os autores (2022).

A opção [2] localizar funcionários visa o gerente consultar os dados dos funcionários existentes na empresa e em breve terá uma atualização afins de melhoria do software, sendo aplicado o consultar por CPF ou Nome do funcionário.

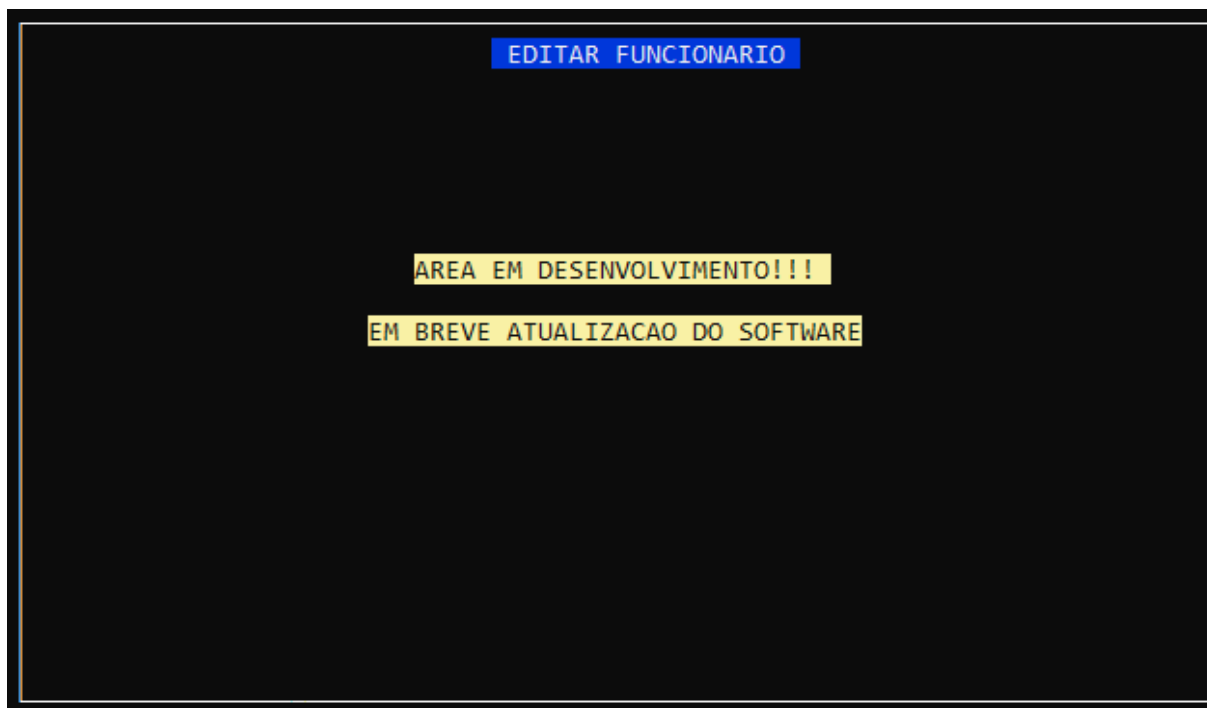
Figura 12 — TELA LOCALIZAR FUNCIONÁRIOS SEM TER O ARQUIVO



Fonte: Os autores (2022).

Nessa tela, ao tentar localizar o funcionario e não possuir cadastro ou o arquivo “funcionario.txt”, apresentará uma mensagem de erro e em seguida voltará ao menu gerente.

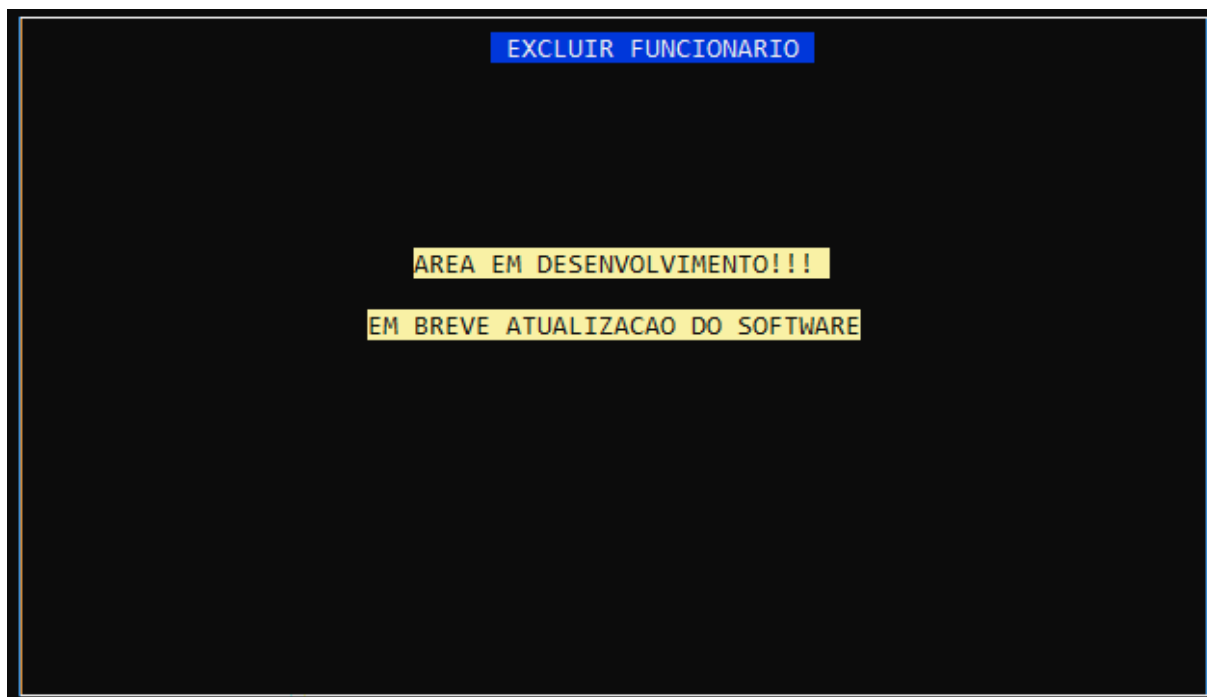
Figura 13 — TELA EDITAR FUNCIONÁRIO



Fonte: Os autores (2022).

A opção [3] editar funcionário visa o gerente atualizar dados dos funcionários existentes na empresa e em breve terá uma atualização afins de melhoria do software, sendo aplicado a consulta por CPF ou nome, e em seguida aparecerá as opções a qual deseja alterar.

Figura 14 — TELA EXCLUIR FUNCIONÁRIO

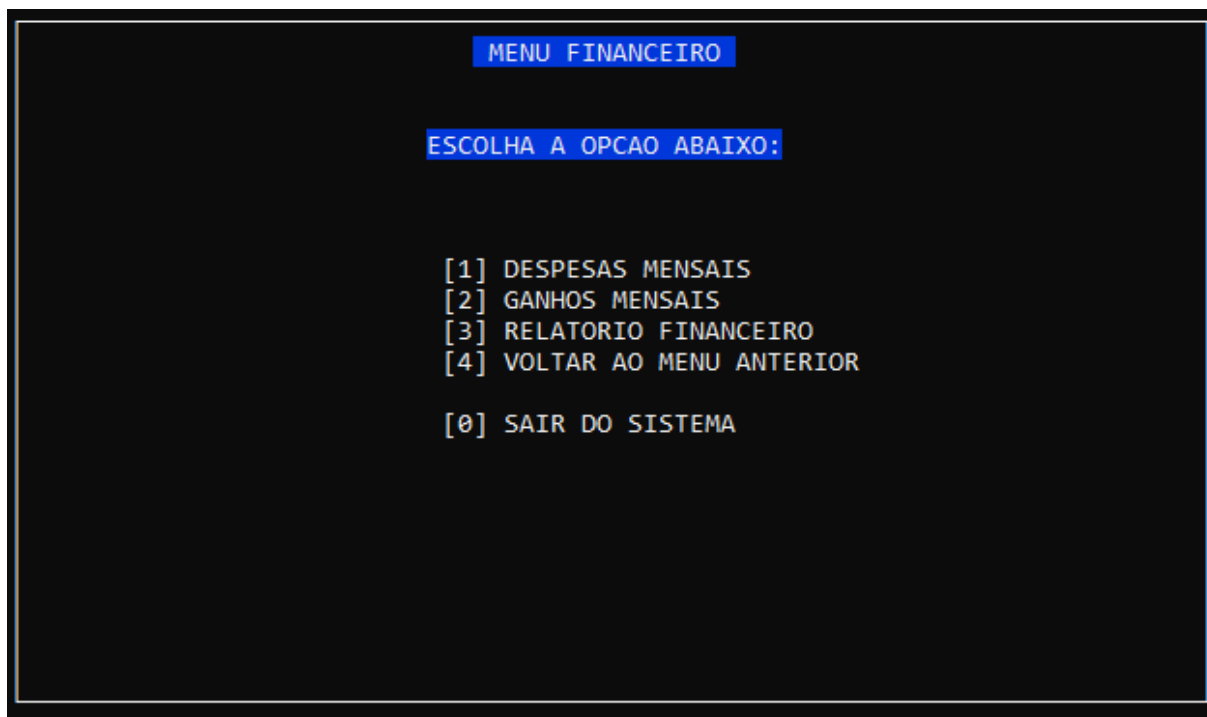


Fonte: Os autores (2022).

A opção [4] excluir funcionário, visa localizar um funcionário por CPF ou Nome, e terá a finalidade de exclusão do funcionário que não faz mais parte da empresa.



Figura 15 — TELA MENU FINANCEIRO



Fonte: Os autores (2022).

Nessa tela, o gerente terá as opções [1] despesas mensais, [2] ganhos mensais, [3] relatório financeiro, [4] voltar ao menu anterior e [0] para sair do sistema.

Figura 16 — TELA DESPESAS MENSAIS

```
DESPESAS MENSAIS

MES/ANO.....:      08/2022
CONTA DE AGUA R$.....:    150.00
CONTA DE LUZ R$.....:    1300.00
CONTA DE INTERNET R$.....:  350.00
ALUGUEL R$.....:    5000.00
PAGAMENTO DE FUNCIONARIOS R$.: 15000.00
DESPESAS EXTRAS R$.....:   375.10

TOTAL BRUTO DE DESPESAS R$...: 22175.10

DESPESAS REGISTRADAS COM SUCESSO!!!
```

Fonte: Os autores (2022).

Nessa tela, o gerente pegará todas as despesas mensais da empresa e lançará no sistema afim de ter um controle de gastos gerados mensais, sendo salvo em um arquivo "despesas.txt".

Figura 17 — TELA RECEITA MENSAIS

The screenshot displays a terminal window titled 'RECEITAS MENSAIS'. It shows the following data:

MES/ANO.....	: 08/2022
SERVICOS R\$.....	: 35000.00
OUTROS R\$.....	: 125.50
TOTAL DE RECEITAS R\$.....	: 35125.50

At the bottom of the screen, a green message box states: 'RECEITAS REGISTRADAS COM SUCESSO!!!'.

Fonte: Os autores (2022).

Nessa tela, o Gerente pegará todas as receitas mensais da empresa e lançará no sistema afim de ter um controle de faturamento mensais gerados pela empresa.

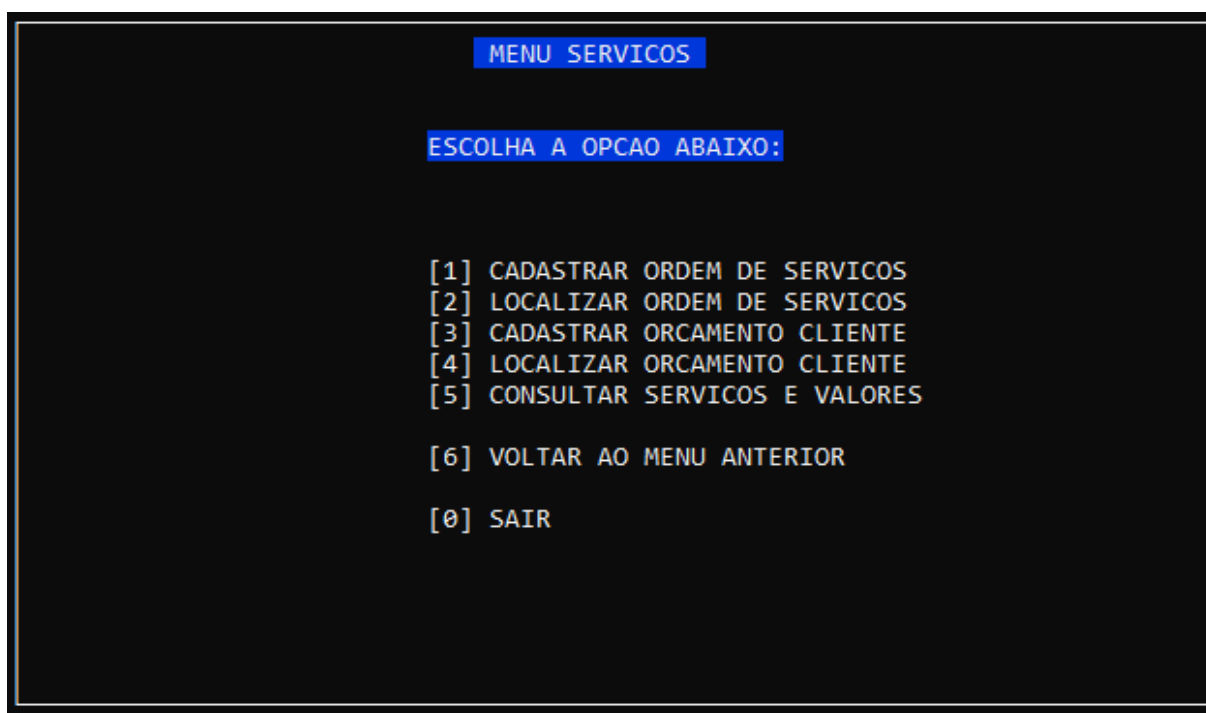
Figura 18 — TELA RELATÓRIO FINANCEIRO



Fonte: Os autores (2022).

Nessa tela, o gerente pegará o valor total de receitas + valor total de despesas e lançará no sistema, afins de conferência, tendo os valores gerados em lucro da empresa.

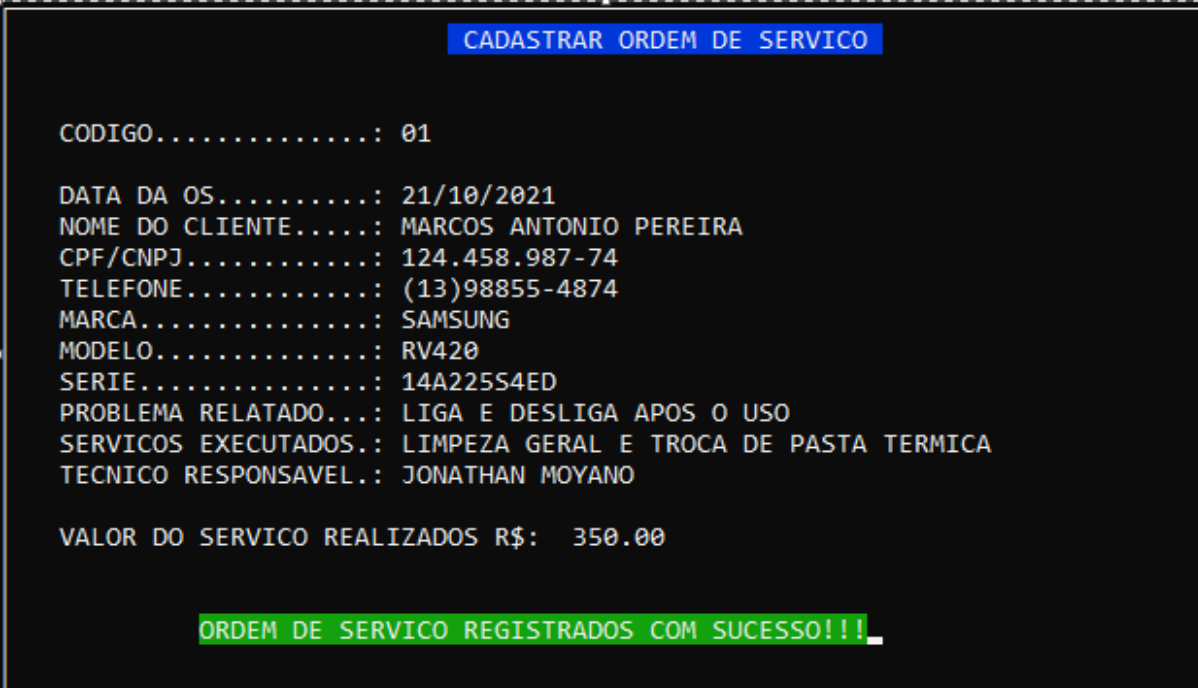
Figura 19 — TELA MENU SERVIÇOS



Fonte: Os autores (2022).

Nessa tela, a recepcionista terá as opções [1] cadastrar ordem de serviços, [2] localizar ordem de serviço, [3] cadastrar orçamentos, [4] localizar orçamento, [5] consultar serviços e valores e [6] para voltar ao menu principal.

Figura 20 — TELA CADASTRAR ORDEM DE SERVIÇOS



The screenshot displays a terminal window with a black background and white text. At the top, a blue header bar contains the text 'CADASTRAR ORDEM DE SERVICO'. Below this, the following information is displayed in a structured format:

```
CODIGO.....: 01  
  
DATA DA OS.....: 21/10/2021  
NOME DO CLIENTE.....: MARCOS ANTONIO PEREIRA  
CPF/CNPJ.....: 124.458.987-74  
TELEFONE.....: (13)98855-4874  
MARCA.....: SAMSUNG  
MODELO.....: RV420  
SERIE.....: 14A225S4ED  
PROBLEMA RELATADO...: LIGA E DESLIGA APOS O USO  
SERVICOS EXECUTADOS.: LIMPEZA GERAL E TROCA DE PASTA TERMICA  
TECNICO RESPONSAVEL.: JONATHAN MOYANO  
  
VALOR DO SERVICO REALIZADOS R$: 350.00  
  
ORDEM DE SERVICO REGISTRADOS COM SUCESSO!!!_
```

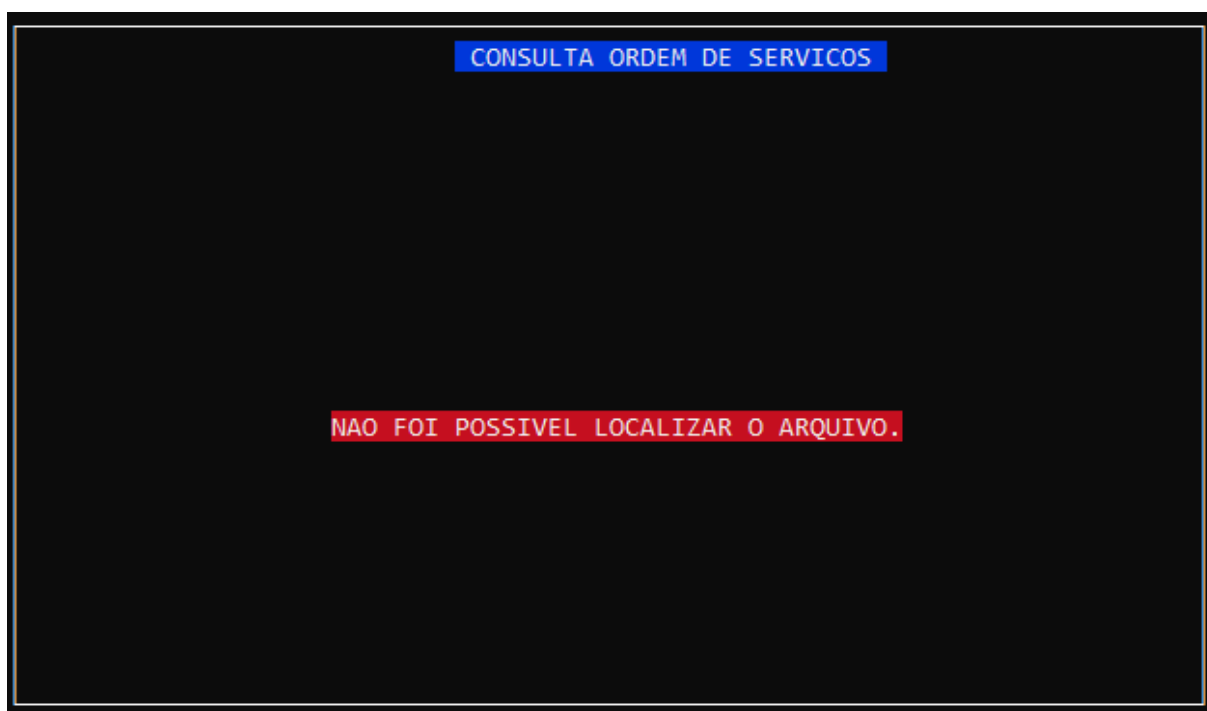
Fonte: Os autores (2022).

Nessa tela, o técnico irá realizar o cadastro de ordem de serviços com base no orçamento feito pela recepcionista, aplicando os serviços feitos e valores, o cadastro ficará salvo em um arquivo chamado “C\_OS.txt” na pasta do arquivo.

Figura 21 — TELA LOCALIZAR ORDEM DE SERVIÇOSOs autores (2022).

Nessa tela, a recepcionista irá realizar a consulta das ordens de serviços criada pelo técnico, afins de repassar as informações ao cliente final.

Figura 21 — TELA LOCALIZAR ORDEM DE SERVIÇO SEM TER O ARQUIVO



Fonte: Os autores (2022).

Nessa tela, ao tentar localizar a ordem de serviço e não possuir o mesmo ou o arquivo "C\_OS.txt", apresentará uma mensagem de erro e em seguida voltará ao menu serviços.

Figura 22 — TELA DE CADASTRO DE ORÇAMENTO

**CADASTRO DE ORÇAMENTO**

CODIGO.....: 01

DATA DO ORÇAMENTO...: 21/10/2021

NOME DO CLIENTE.....: MARCOS ANTONIO PEREIRA

CPF/CNPJ.....: 124.458.987-74

TELEFONE.....: (13)98855-4874

MARCA.....: SAMGUNG

MODELO.....: RV420

SERIE.....: 14A225S4ED

POSSUI BATERIA..... SIM

DEFEITO RELATADO....: LIGA E DESLIGA APOS O USO

**ORÇAMENTO REGISTRADO COM SUCESSO!!!**

Fonte: Os autores (2022).

Nessa tela, a recepcionista irá realizar o cadastro de orçamentos com base nas informações que o cliente irá repassar, o cadastro ficará salvo em um arquivo "orçamento.txt" na pasta do arquivo.



Figura 23 — TELA LOCALIZAR ORÇAMENTO



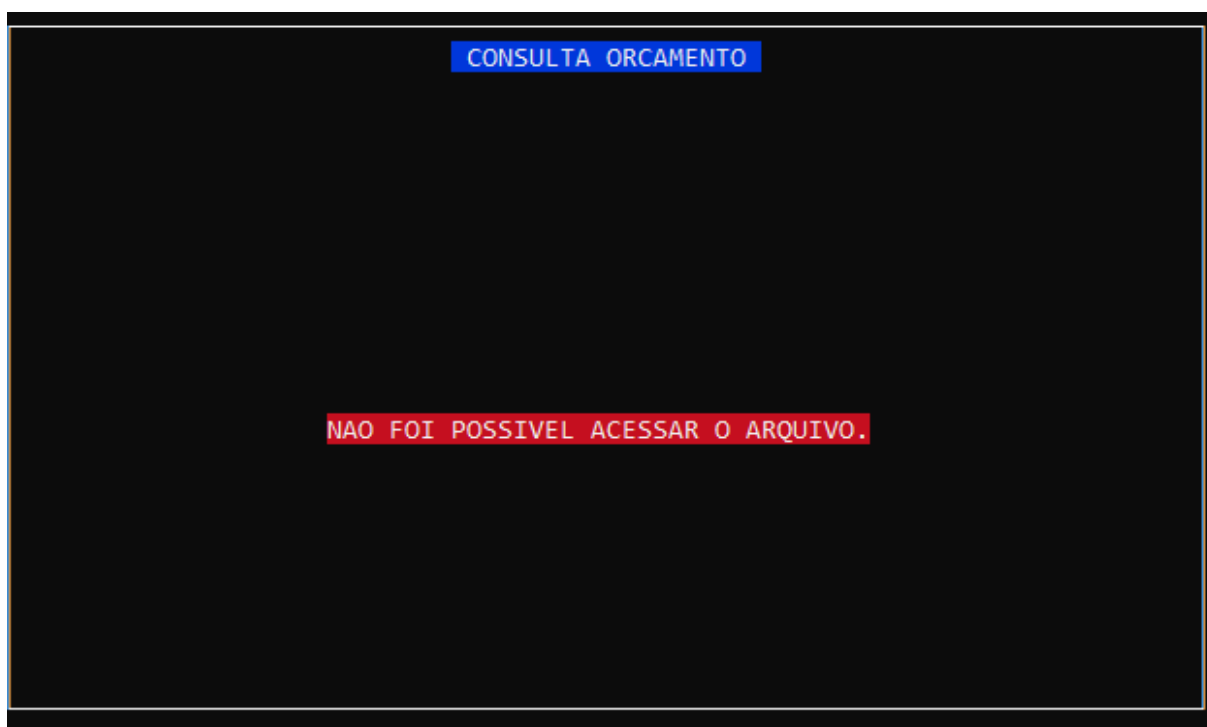
A screenshot of a terminal window with a black background and white text. At the top center, there is a blue rectangular button with the text "CONSULTA ORÇAMENTO" in white. Below the button, the following information is displayed in a monospaced font:

```
CODIGO.....:1
DATA DO ORÇAMENTO...:21/10/2021
NOME DO CLIENTE.....:MARCOS ANTONIO PEREIRA
CPF/CNPJ.....:124.458.987-74
TELEFONE.....:(13)98855-4874
MARCA.....:SAMGUNG
MODELO.....:RV420
SERIE.....:14A225S4ED
POSSUI BATERIA.....:SIM
DEFEITO RELATADO....:LIGA E DESLIGA APOS O USO
```

Fonte: Os autores (2022).

Nessa tela, o técnico irá realizar a consulta dos orçamentos conforme cadastrados pela recepcionista, tendo as informações e defeito do aparelho.

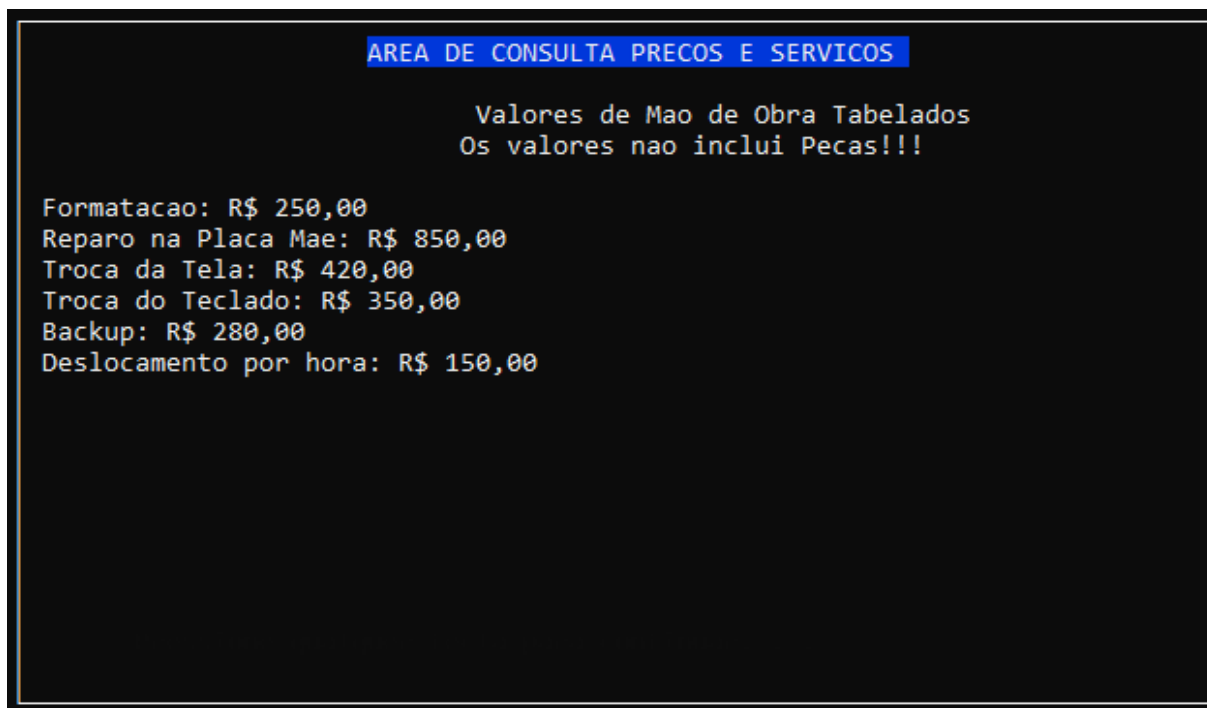
Figura 24 — TELA LOCALIZAR ORÇAMENTO SEM TER O ARQUIVO



Fonte: Os autores (2022).

Nessa tela, ao tentar localizar o orçamento e não possuir o mesmo ou o arquivo "orçamento.txt", apresentará uma mensagem de erro e em seguida voltará ao menu serviços.

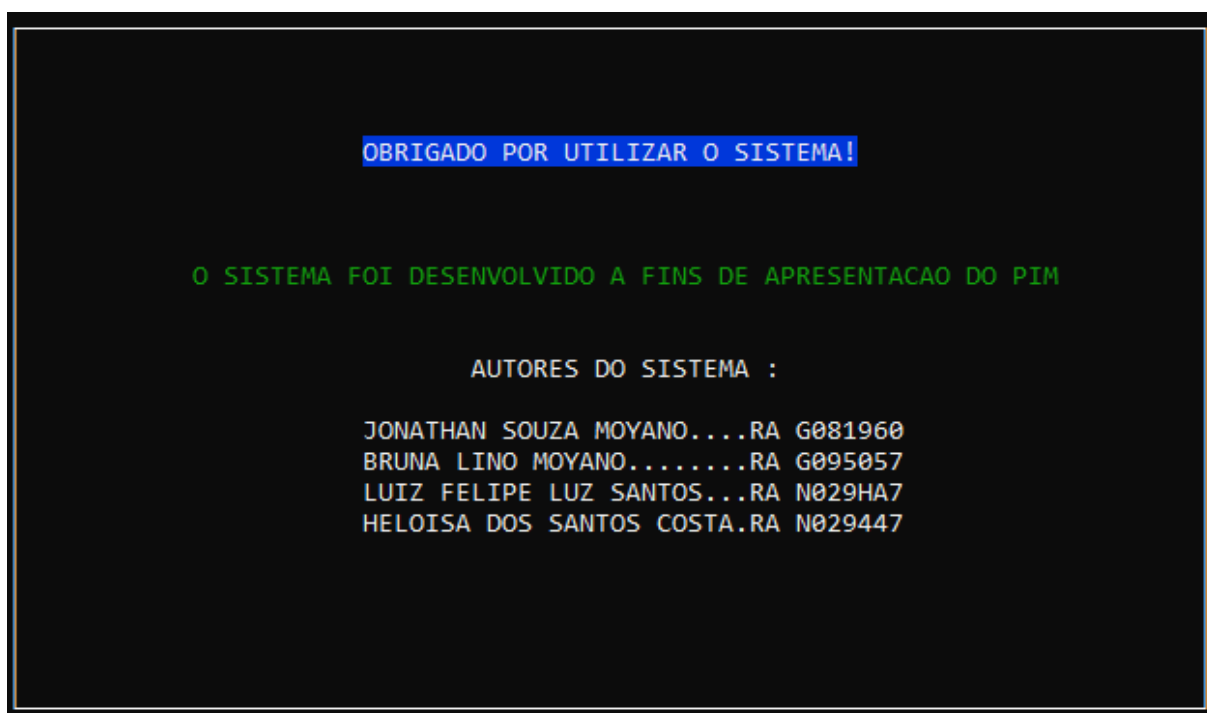
Figura 25 — TELA DE CONSULTA PREÇOS E SERVIÇOS



Fonte: Os autores (2022).

Nessa tela, a recepcionista irá realizar a consulta de serviços e valores afins de passar a estimativa de preço ao cliente e o técnico irá consultar para aplicar na ordem de serviços os valores dos serviços realizados pelo mesmo.

Figura 26 — TELA SAIR DO SISTEMA



Fonte: Os autores (2022).

Nessa tela, irá aparecer uma mensagem de agradecimento por ter utilizado o sistema e também, os autores que realizaram a criação do sistema.

### 7.3 LINHA DE CODIGO DO SISTEMA CTECH

```

//BIBLIOTECAS
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include <conio.h>
#include <time.h>
#include <locale.h>
#include <Windows.h>
#include <unistd.h>
#define SIZE 200
//constantes com os nomes e codigos das cores padrao
#define PRETO 0
#define AZUL 1
#define VERDE 2
#define CIANO 3
  
```

```

#define VERMELHO 4
#define MAGENTA 5
#define MARROM 6
#define CINZA CLARO 7
#define CINZA ESCURO 8
#define AZUL CLARO 9
#define VERDE CLARO 10
#define CIANO CLARO 11
#define VERMELHO CLARO 12
#define MAGENTA CLARO 13
#define AMARELO 14
#define BRANCO 15
#define DBF rewind(stdin)
/*/

#####
##### LEGENDA #####
#####

DOUBLE = VARIÁVEL PARA NÚMEROS REAIS
INT = VARIÁVEL PARA INTEIRO
CHAR = VARIÁVEL PARA CARACTERE
STRUCT = VARIÁVEIS
SCANF = LEITURA DE DADOS BÁSICOS - SEM ESPAÇAMENTO
PRINTF = MENSAGEM NA TELA
FPRINTF = PERMITE ESCRITA DE DADOS NO ARQUIVO;
FGETS = LEITURA DE TEXTOS ATÉ QUEBRA DE LINHA;
IF = SE
ELSE = SENÃO
WHILE = SE FOR VERDADEIRO EXECUTA E VOLTA, SE FOR FALSO
PULA FORA;
SWITCH = TESTA IGUALDADE - DECISÃO
GOTO = IR PARA - POSICIONAMENTO DO CURSOR
FFLUSH (STDIN) = LIMPAR O BUFFER DE SAÍDA DE DADOS ( LIMPAR
MEMÓRIA )
BREAK = PARAR DE EXECUTAR A REPETIÇÃO
CASE = ESTRUTURA DE ESCOLHA
SYSTEM ("CLS") = LIMPAR TELA
SYSTEM ("PAUSE") = PAUSAR A TELA
STRCPN = RECEPTOR DE 2 STRINGS COMO ENTRADA DE DADOS
// = COMENTAR NA LINHA DE CÓDIGO;

```

SLEEP () = LOOPING NA TELA  
 FILE \* = ARQUIVO  
 FCLOSE = FECHAR O ARQUIVO;  
 FOPEN = ABRIR O ARQUIVO;  
 W = SE NAO HOUVER ARQUIVO ELE CRIA, CASO EXISTA, ELE  
 SUBSTITUI O DIGITADO (ESCRITA)  
 AB+ = SE NAO HOUVER ARQUIVO ELE CRIA, CASO EXISTA, ELE PULA  
 LINHA E ACRESCENTA (ESCRITA E LEITURA)  
 .TXT = EXTENSAO DO ARQUIVO QUE VAI SER SALVO (BLOCO DE  
 NOTAS)  
 .XLSX EXTENSAO DO ARQUIVO QUE VAI SER SALVO (EXCEL)  
 VOID =  
 /\*/  
 // GLOBAIS  
 // CADASTRO DE ORDEM DE SERVIÇO  
 int codigoOS[SIZE];  
 char Marca\_OS[SIZE][12];  
 char Modelo\_OS[SIZE] [10];  
 char Numero\_de\_serie\_OS[SIZE][10];  
 char Defeito\_Relatado\_OS[SIZE][200];  
 char CPF\_CNPJ\_OS[SIZE][50];  
 char Nome\_Completo\_OS[SIZE][100];  
 char Data\_OS[SIZE] [10];  
 char Telefone\_OS[SIZE][50];  
 char Estado\_OS[SIZE][10];  
 char Cidade\_OS[SIZE][10];  
 char hora\_OS[SIZE][10];  
 char Servicos\_Executado\_OS[SIZE][200];  
 char Tecnico\_Responsavel\_OS[SIZE][100];  
 // CADASTRO DE CLIENTES  
 int C\_Cliente[SIZE];  
 char nome\_Cliente[SIZE][100];  
 char endereco\_Cliente [SIZE][200];  
 char telefone\_Cliente[SIZE][50];  
 char estado\_Cliente[SIZE] [50];  
 char cpf\_Cliente[SIZE][50];  
 char cep\_Cliente[SIZE][50];  
 int codigo\_Cliente=0;  
 char data\_nasc\_Cliente [SIZE][20];

```

char cidade_Cliente [SIZE][20];
// CADASTRO DE FUNCIONARIO
char nome_Funcionario[SIZE][100];
char endereco_Funcionario[SIZE][100];
char telefone_Funcionario[SIZE][50];
char rg_Funcionario[SIZE][50];
char cpf_Funcionario[SIZE][50];
char funcao_Funcionario[SIZE][50];
char data_admissao_Funcionario[SIZE][20];
int codigo_Funcionario[SIZE];
char data_nasc_Funcionario[SIZE][20];
char estado_Funcionario[SIZE][50];
char cidade_Funcionario[SIZE][20];
char cep_Funcionario[SIZE][50];
// CADASTRO DE ORCAMENTO
int codigoOrcamento[SIZE];
char Data_Orcamento [SIZE][20];
char Marca_Orcamento[SIZE][50];
char Modelo_Orcamento [SIZE][50];
char Numero_de_serie_Orcamento[SIZE] [20];
char Defeito_Relatado_Orcamento[SIZE] [100];
char Possui_arranhos_Orcamento[SIZE] [10];
char Possui_Bateria_Orcamento [SIZE][10];
char Possui_fonte_Orcamento [SIZE][10];
char nome_do_cliente_Orcamento[SIZE][100];
char telefone_Orcamento[SIZE];
int CPF_CNPJ_Orcamento[SIZE];
// SERVIÇOS
double Formatacao[SIZE][10];
double Reparo_Placa_Mae[SIZE][10];
double Troca_de_Tela[SIZE][10];
double Remocao_de_Virus[SIZE][10];
double Troca_de_Teclado[SIZE][10];
double Backup[SIZE][10];
double Upgrade[SIZE][10];
double Deslocamento[SIZE][10];
double preco[SIZE][10];
//RECEITAS

```

```

int C_Receita[SIZE];
double Servicos[SIZE][10];
double Outros[SIZE][10];
//DESPESAS
int C_Despesas[SIZE];
double Conta_agua[SIZE][10];
double Conta_luz[SIZE][10];
double Conta_internet[SIZE][10];
double Conta_aluguel[SIZE][10];
double pgto_funcionario[SIZE][10];
double Conta_extras[SIZE][10];
double total_despesas[SIZE][10];
char data_despesa[SIZE];
int total[SIZE];
char data_receita[SIZE][50];
//ESTRUTURA
struct Servicos
{
    char C_Servicos[10];
    double Formatacao[10];
    double Reparo_Placa_Mae[10];
    double Troca_de_Tela[10];
    double Remocao_de_Virus[10];
    double Troca_de_Teclado[10];
    double Backup[10];
    double Upgrade[10];
    double Deslocamento[10];
    double preco[10];
    char nome_do_cliente[50];
};
struct incluir_Orcamento
{
    char C_Orcamento[10];
    float codigoOrcamento;
    char Data_Orcamento [20];
    char Marca_Orcamento[50];
    char Modelo_Orcamento [50];
    char Numero_de_serie_Orcamento [20];

```



```

    char Defeito_Relatado_Orcamento [200];
    char Possui_arranhoes_Orcamento [10];
    char Possui_Bateria_Orcamento [10];
    char Possui_fonte_Orcamento [10];
    char nome_do_cliente_Orcamento[100];
    char telefone_Orcamento[20];
    char CPF_CNPJ_Orcamento[50];
};

struct incluirOS
{
    char C_OS[10];
    char codigoOS[100];
    char Marca_OS[12];
    char Modelo_OS [10];
    char Numero_de_serie_OS [10];
    char Defeito_Relatado_OS [200];
    char CPF_CNPJ_OS[20];
    char Nome_Completo_OS[100];
    char Data_OS [10];
    char Telefone_OS[20];
    char Estado_OS[2];
    char Cidade_OS[10];
    char hora_OS[10];
    char Servicos_Executado_OS[200];
    char Tecnico_Responsavel_OS[100];
    float valor_OS;
};

struct incluir_Cliente
{
    int C_Cliente;
    char nome_Cliente[100];
    char endereco_Cliente[200];
    char telefone_Cliente[50];
    char estado_Cliente[50];
    char cpf_Cliente[14];
    char cep_Cliente[14];
    int codigo_Cliente;
    char data_nasc_Cliente[20];

```

```

    char cidade_Cliente[20];
};
struct incluir_funcionarios
{
    int C_Funcionarios[SIZE];
    char nome_Funcionario [SIZE][100];
    char endereco_Funcionario [SIZE][200];
    char telefone_Funcionario[SIZE] [50];
    char rg_Funcionario[SIZE][12];
    char cpf_Funcionario [SIZE][14];
    char funcao_Funcionario[SIZE][50];
    char data_admissao_Funcionario[SIZE][20];
    int codigo_Funcionario[SIZE];
    char data_nasc_Funcionario[SIZE][20];
    char estado_Funcionario[SIZE][50];
    char cidade_Funcionario[SIZE][20];
    char cep_Funcionario[SIZE][10];
};
struct Receita
{
    int C_Receita[SIZE];
    double Servicos[SIZE];
    double Outros[SIZE];
    double total_receitas[SIZE];
};
struct Despesas
{
    char C_Despesas[SIZE][10];
    double Conta_agua[SIZE];
    double Conta_luz[SIZE];
    double Conta_internet[SIZE];
    double Conta_aluguel[SIZE];
    double pgto_funcionario[SIZE];
    double Conta_extras[SIZE];
    double total_despesas[SIZE];
    char data_despesa[SIZE];
};
struct RelatorioFinanceiro

```

```

{
    int C_Despesas;
    double Conta_agua;
    double Conta_luz;
    double Conta_internet;
    double Conta_aluguel;
    double pgto_funcionario;
    double Conta_extras;
    double total_despesas;
    int C_Receita;
    double Servicos;
    double Outros;
    char total[1000];
    char data_despesa;
};

void mudacor(int corTexto, int corFundo)
{
    HANDLE tela;
    tela = GetStdHandle(STD_OUTPUT_HANDLE);
    int novasCores;
    novasCores = corFundo;
    novasCores = _rotl(novasCores,4) | corTexto;
    SetConsoleTextAttribute(tela, novasCores);
}

//MARGENS
void gotoxy( int x, int y );
void bordas();
//TELA DE LOGIN
void telaLogin();
void telaLogin2();
//MENU
void menuGeral();
void menuGerente();
void menuFinanceiro();
void menuServicos();
void despesas();
void menuFuncionario();
void Receita();

```

```

//MENU CADASTRO
void menuCliente();
void excluirCliente();
void Receita();
//CADASTROS
void incluirCliente();
void incluirfuncionarios();
void incluirServicos();
void Cadastrar_OS();
void incluir_Orcamento();
void editarfuncionario();
void excluirfuncionario();
//EDITAR CLIENTES
void editarCliente();
void pesquisa();
//CONSULTAS
void consultaServicos();
void consultaOS();
void consultaFuncionario();
void consultaOrcamento();
//CATALAGO
void catalogoServico();
//RELATORIO
void relatorio_de_OS();
void relatorio_de_Servicos();
void relatorio_Financeiro();
void menuFinanceiro();
//SAIR
void sair();
int main ()
{
    system("mode con:cols=80 lines=25");
    telaLogin();
    telaLogin2();
    catalogoServico();
    menuServicos();
    system("title CTECH - Sistema de Manutenção em Notebooks");
//FUNCAO MENU
    menuGeral();

```

```

    return 0;
}
void gotoxy( int x, int y )
{
    printf("%c[%d;%df", 0x1B,y,x );
}
void bordas ()
{
    for (int x =3; x <79; x++ )
    {
        gotoxy (x,2); // POSICIONAR COLUNA E LINHA
        printf ("%c",196); //BORDA HORIZONTAL SUPERIOR
        gotoxy (x,24); // POSICIONAR COLUNA E LINHA
        printf ("%c",196); // BORDA HORIZONTAL INFERIOR
    }
    for (int y =3; y <24; y++ )
    {
        gotoxy (2,2); // POSICIONAR COLUNA E LINHA
        printf ("%c",218); //CANTO SUPERIOR ESQUERDO
        gotoxy (79,2); // POSICIONAR COLUNA E LINHA
        printf ("%c",191); //CANTO SUPERIOR DIREITO
        gotoxy (2,y); // POSICIONAR COLUNA E LINHA
        printf ("%c", 179); //BORDA VERTICAL DO LADO DIREITO
        gotoxy (79,y); // POSICIONAR COLUNA E LINHA
        printf ("%c",179); //BORDA VERTICAL DO LADO ESQUERDO
        gotoxy (2,24); // POSICIONAR COLUNA E LINHA
        printf ("%c",192); //CANTO INFERIOR ESQUERDO
        gotoxy (79,24); // POSICIONAR COLUNA E LINHA
        printf ("%c",217); //CANTO INFERIOR DIREITO
    }
}
}
void telaLogin()
{
    struct Tela_Login_Principal;
    int ch;
    int i = 0;
    char login [20] = "ctech"; // LOGIN DO USUARIO
    char senha [20] = "123"; // SENHA DO USUARIO
    char login10 [20] = "admin"; // LOGIN DO USUARIO

```

```

char senha10 [20] = "admin";
char login1[20] = {0};
char senha1[20] = {0};
system ("cls"); // LIMPAR TELA
system ("color 0F"); // ALTERAR A COR DA TELA PRETA COM VERDE
bordas (); // APLICANDO BORDA NO MENU LOGIN PRINCIPAL
gotoxy (25,3); // POSICIONAR COLUMNA E LINHA
mudacor (BRANCO,AZUL); // ALTERAR COR DA LINHA
printf (" CTECH ");// NOME DA EMPRESA
gotoxy (25,4); // POSICIONAR COLUMNA E LINHA
printf (" Tecnologia personalizada ");// SLOGAN
gotoxy (34,8); // POSICIONAR COLUMNA E LINHA
mudacor(BRANCO,AZUL); // ALTERAR COR DA LINHA
printf (" TELA DE LOGIN "); // CABECALHO
mudacor(BRANCO,PRETO); // ALTERAR COR DA LINHA
gotoxy (25,13); // POSICIONAR COLUMNA E LINHA
printf ("DIGITE O USUARIO: "); // MENSAGEM FIXA
gotoxy (25,14); // POSICIONAR COLUMNA E LINHA
printf ("DIGITE A SENHA..: "); // MENSAGEM FIXA
gotoxy (45,13); // POSICIONAR COLUMNA E LINHA
scanf ("%s",login1); // LEITURA DO LOGIN
fflush (stdin); // LIMPAR BUFFER DO TECLADO
gotoxy (45,14); // POSICIONAR COLUMNA E LINHA
while (ch !=13)
{
    ch = getch();
    if (ch == 13)
        break ;
    fflush (stdin);
    if (ch == 8)
    {
        if(i==0)
            continue;
        --i;
        for(int a=0; a<i; a++)
            printf("*");
        continue;
    }
    printf("*");
}

```

```

        senha1[i] = ch;
        i++;
    }
    if((strcmp(login, login1)== 0 && strcmp(senha, senha1)== 0)) // VALIDACAO
DO LOGIN E SENHA
    {
        mudacor(BRANCO,VERDE); // ALTERAR COR DA TELA
        gotoxy(25,19); // POSICIONAR COLUNA E LINHA
        printf(" LOGADO COMO FUNCIONARIO !!!"); // MENSAGEM NA TELA
        sleep (1); // LOOPING DE 1 SEGUNDO
        mudacor(BRANCO,PRETO); // ALTERAR COR DA LINHA
        menuGeral(); // DIRECIONAR AO MENU PRINCIPAL
    }
    else if ((strcmp(login10, login1)== 0 && strcmp(senha10, senha1)== 0)) //
CASO CONTRARIO
    {
        mudacor(BRANCO,VERDE); // ALTERAR COR DA TELA
        gotoxy(25,19); // POSICIONAR COLUNA E LINHA
        printf(" LOGADO COMO GERENTE !!!");
        sleep (1); // LOOPING DE 1 SEGUNDO
        mudacor(BRANCO,PRETO); // ALTERAR COR DA LINHA
        menuGerente();
    }
    else
    {
        mudacor(BRANCO,VERMELHO);
        gotoxy (20,19); // POSICIONAR COLUNA E LINHA
        printf("LOGIN E/OU SENHA INVALIDOS, TENTE NOVAMENTE!"); //
MENSAGEM DE FALHA NA AUTENTICACAO
        sleep(1); // LOOPING DE 1 SEGUNDO
        mudacor(BRANCO,PRETO); // ALTERAR COR DA LINHA
        telaLogin(); // DIRECIONAR A TELA DE LOGIN NOVAMENTE
    }
}
void telaLogin2()
{
    struct Tela_Login_Gerente;
    int ch2;
    int i2 = 0;

```

```

char loginUser[20] = "admin"; // LOGIN DO GERENTE
char login2[20] = {0};
char senhaUser[20] = "admin"; // SENHA DO GERENTE
char senha2[20] = {0};
system ("cls"); // LIMPAR TELA
system ("color 0F"); // ALTERAR A COR DA TELA PRETA COM VERDE
bordas (); //APLICANDO BORDA NO MENU LOGIN GERENTE
mudacor(BRANCO,AZUL); // ALTERAR COR DA LINHA
gotoxy (25,3); // POSICIONAR COLUMNA E LINHA
printf (" CTECH "); // NOME DA EMPRESA
gotoxy (25,4); // POSICIONAR COLUMNA E LINHA
printf (" Tecnologia personalizada "); // SLOGAN
gotoxy (33,8); // ALTERAR COR DA LINHA
mudacor(BRANCO,AZUL); // POSICIONAR COLUMNA E LINHA
printf (" TELA DE LOGIN "); // CABECALHO
gotoxy(33,9); // POSICIONAR COLUMNA E LINHA
mudacor(VERMELHO,AZUL); // ALTERAR COR DA LINHA
printf(" ACESSO GERENTE "); // TITULO
mudacor(BRANCO,PRETO); // ALTERAR COR DA LINHA
gotoxy (25,13); // POSICIONAR COLUMNA E LINHA
printf ("DIGITE O USUARIO: "); // MENSAGEM FIXA
gotoxy (25,14); // POSICIONAR COLUMNA E LINHA
printf ("DIGITE A SENHA..: "); // MENSAGEM FIXA
gotoxy (45,13); // POSICIONAR COLUMNA E LINHA
scanf ("%s",login2); // LEITURA DO LOGIN
fflush (stdin); // LIMPAR BUFFEN DE TECLADO
gotoxy (45,14); // POSICIONAR COLUMNA E LINHA
while (ch2!=13)
{
    ch2 = getch();
    if (ch2 == 13)
        break ;
    if (ch2 == 8)
    {
        if(i2 == 0) continue;
        --i2;
        for(int a = 0; a < i2; a++)
            printf("*");
        continue;
    }
}

```



```

    }
    printf("*");
    senha2[i2] = ch2;
    i2++;
}
if(strcmp(loginUser,login2) == 0 && strcmp(senhaUser,senha2)== 0) //
VALIDACAO DO LOGIN E SENHA
{
    mudacor(BRANCO,VERDE); // ALTERAR COR DA TELA VERDE COM
BRANCO
    gotoxy(20,19); // POSICIONAR COLUNA E LINHA
    printf("LOGADO COM SUCESSO !!!POR FAVOR, AGUARDE..."); //
MENSAGEM NA TELA
    sleep(1); // LOOPING NO SISTEMA DE 1 SEGUNDO
    mudacor(BRANCO,PRETO); // ALTERAR COR DA TELA PRETO COM
AZUL
    menuGerente(); // APOS LOGAR, DIRECIONANDO AO MENU
GERENTE
}
else // CASO CONTRARIO
{
    mudacor(BRANCO,VERMELHO); // ALTERAR COR DA TELA PARA
VERMELHO
    gotoxy(21,20); // POSICIONAR COLUNA E LINHA
    printf("LOGIN E SENHA INVALIDOS, TENDE NOVAMENTE!"); //
MENSAGEM DE ERRO NA AUTENTICACAO
    sleep(1); // LOOPING NO SISTEMA DE 1 SEGUNDO
    mudacor(BRANCO,PRETO); // ALTERAR COR DA TELA PARA PRETO
COM AZUL
    menuGeral(); // DIRECIONANDO PARA VOLTAR AO MENU PRINCIPAL
}
}
void menuGeral()
{
    system("color 0F"); // ALTERAR A COR DA TELA PARA AZUL E BRANCO
    system("cls"); // LIMPAR TELA
    bordas (); // APLICANDO BORDAS NO MENU PRINCIPAL
    char opcao2; //VARIABEL
    mudacor(BRANCO,AZUL); // MUDAR COR DA LINHA

```

```

gotoxy (32,3); // POSICIONAR COLUNA E LINHA
printf ("MENU PRINCIPAL"); // MENSAGEM FIXA
mudacor(BRANCO,AZUL); // MUDAR COR DA LINHA
gotoxy (29,6); // POSICIONAR COLUNA E LINHA
printf ("ESCOLHA A OPCAO ABAIXO:"); // MENSAGEM FIXA
mudacor(BRANCO,PRETO); // MUDAR COR DA LINHA
gotoxy (32,10); // POSICIONAR COLUNA E LINHA
printf ("[1] MENU CLIENTE"); // MENSAGEM FIXA
gotoxy (32,11); // POSICIONAR COLUNA E LINHA
printf ("[2] MENU GERENTE"); // MENSAGEM FIXA
gotoxy (32,12); // POSICIONAR COLUNA E LINHA
printf ("[3] MENU SERVICOS"); // MENSAGEM FIXA
gotoxy (32,14); // POSICIONAR COLUNA E LINHA
printf ("[0] SAIR DO SISTEMA"); // MENSAGEM FIXA
gotoxy (53,6); // POSICIONAR COLUNA E LINHA
mudacor(PRETO,PRETO); // MUDAR COR DA LINHA
    opcao2 = getche(); // LER COMO CARACTERE E CONVERTE PARA
INTEIRO
switch (opcao2) // ESCOLHA DE DECISAO
{
case '1': // SE APERTAR A OPCAO 1
    menuCliente(); // DIRECIONANDO PARA MENU CLIENTE
    fflush(stdin); // LIMPAR O BUFFEN DO TECLADO
    break; // PARAR
case '2': // SE APERTAR A OPCAO 2
    telaLogin2(); // DIRECIONANDO PARA TELA DE LOGIN DO GERENTE
    fflush(stdin); // LIMPAR O BUFFEN DO TECLADO
    break; // PARAR
case '3': // SE APERTAR A OPCAO 3
    menuServicos(); // DIRECIONANDO PARA CONSULTA DE PRECOS E
SERVICOS
    fflush(stdin); // LIMPAR O BUFFEN DO TECLADO
    break; // PARAR
case '0':
    sair(); // SE APERTAR A OPCAO 0
    fflush(stdin); // LIMPAR O BUFFEN DO TECLADO
    break; // PARAR
default: // SE NENHUMA OPCAO DIGITADA

```

```

    fflush(stdin);
    gotoxy (31,20); // POSIONAR COLUNA E LINHA
    mudacor(BRANCO,VERMELHO); // MUDAR COR DA LINHA
    printf("OPCAO INVALIDA!!!\n\n"); // MENSAGEM FIXA
    sleep (1); // LOPPING 1 SEGUNDO
    mudacor(BRANCO,PRETO); // MUDAR COR DA LINHA
    system("cls"); // LIMPAR TELA
    menuGeral(); // DIRECIONAMENTO PARA O MENU GERAL
    break; // PARAR DE VARRER
};
}
void menuCliente()
{
    system("color 0F"); // ALTERAR A COR DA TELA PARA AZUL E BRANCO
    system("cls"); // LIMPAR TELA
    bordas (); // APLICANDO BORDAS NO MENU CLIENTE
    char opcao2; // VARIABEL
    mudacor(BRANCO,AZUL); // MUDAR COR DA LINHA
    gotoxy (33,3); // POSICIONAR COLUNA E LINHA
    printf ( " MENU CLIENTE "); // MENSAGEM FIXA
    mudacor(BRANCO,AZUL); // MUDAR COR DA LINHA
    gotoxy (29,6); // POSICIONAR COLUNA E LINHA
    printf ("ESCOLHA A OPCA O ABAIXO:"); // MENSAGEM FIXA
    mudacor(BRANCO,PRETO); // MUDAR COR DA LINHA
    gotoxy (32,10); // POSICIONAR COLUNA E LINHA
    printf ("[1] CADASTRAR CLIENTE"); // MENSAGEM FIXA
    gotoxy (32,11); // POSICIONAR COLUNA E LINHA
    printf ("[2] LOCALIZAR CLIENTE"); // MENSAGEM FIXA
    gotoxy (32,12); // POSICIONAR COLUNA E LINHA
    printf ("[3] EDITAR CLIENTE"); // MENSAGEM FIXA
    gotoxy (32,13); // POSICIONAR COLUNA E LINHA
    printf ("[4] EXCLUIR CLIENTE"); // MENSAGEM FIXA
    gotoxy (32,14); // POSICIONAR COLUNA E LINHA
    printf ("[5] VOLTAR AO MENU PRINCIPAL");// MENSAGEM FIXA
    gotoxy (32,15); // POSICIONAR COLUNA E LINHA
    printf ("[0] SAIR DO SISTEMA"); // MENSAGEM FIXA
    gotoxy (53,6); // POSICIONAR COLUNA E LINHA
    mudacor(PRETO,PRETO); // MUDAR COR DA LINHA

```

```

        opcao2 = getche(); // LER COMO CARACTERE E CONVERTE PARA
INTEIRO
        switch (opcao2) // ESCOLHA DE DECISAO
        {
        case '1': // SE APERTAR A OPCAO 1
            incluirCliente(); // DIRECIONANDO PARA CADASTRAR CLIENTE
            fflush(stdin); // LIMPAR O BUFFEN DO TECLADO
            break; // PARAR
        case '2': // SE APERTAR A OPCAO 2
            consultaCliente(); // DIRECIONANDO PARA CONSULTAR CLIENTE
            fflush(stdin); // LIMPAR O BUFFEN DO TECLADO
            break; // PARAR
        case '3': // SE APERTAR A OPCAO 3
            editarCliente(); // DIRECIONANDO PARA CADASTRAR ORDEM DE
SERVICOS
            fflush(stdin); // LIMPAR O BUFFEN DO TECLADO
            break;
        case '4': // SE APERTAR A OPCAO 3
            excluirCliente(); // DIRECIONANDO PARA CADASTRAR ORDEM DE
SERVICOS
            fflush(stdin); // LIMPAR O BUFFEN DO TECLADO
            break;
        case '5': // SE APERTAR A OPCAO 3
            menuGeral(); // DIRECIONANDO PARA CADASTRAR ORDEM DE
SERVICOS
            fflush(stdin); // LIMPAR O BUFFEN DO TECLADO
            break; // PARAR
        case '0': // SE APERTAR A OPCAO 0
            sair(); // DIRECIONANDO PARA SAIR DO SISTEMA
            fflush(stdin); // LIMPAR O BUFFEN DO TECLADO
            break; // PARAR
        default: // SE NENHUMA OPCAO DIGITADA
            fflush(stdin); // LIMPAR BUFFER DO TECLADO
            gotoxy (31,20); // POSICIONAMENTO COLUNA E LINHA
            mudacor(BRANCO,VERMELHO); // MUDAR COR DA LINHA
            printf("OPCAO INVALIDA!!!\n\n"); // MENSAGEM FIXA
            sleep (1); // LOOPING DE 1 SEGUNDO
            mudacor(BRANCO,PRETO); // MUDAR COR DA LINHA

```

```

        getch();
        menuCliente(); // DIRECIONAMENTO PARA O MENU CLIENTE
    }
}

void menuFuncionario()
{
    system("color 0F"); // ALTERAR A COR DA TELA PARA AZUL E BRANCO
    system("cls"); // LIMPAR TELA
    bordas (); // APLICANDO BORDAS NO MENU CLIENTE
    char opcao2; // VARIÁVEL
    mudacor(BRANCO,AZUL); // MUDAR COR DA LINHA
    gotoxy (32,3); // POSICIONAR COLUNA E LINHA
    printf ( " MENU FUNCIONARIO "); // MENSAGEM FIXA
    mudacor(BRANCO,AZUL); // MUDAR COR DA LINHA
    gotoxy (29,6); // POSICIONAR COLUNA E LINHA
    printf ("ESCOLHA A OPCAO ABAIXO:"); // MENSAGEM FIXA
    mudacor(BRANCO,PRETO); // MUDAR COR DA LINHA
    gotoxy (32,10); // POSICIONAR COLUNA E LINHA
    printf ("[1] CADASTRAR FUNCIONARIO"); // MENSAGEM FIXA
    gotoxy (32,11); // POSICIONAR COLUNA E LINHA
    printf ("[2] LOCALIZAR FUNCIONARIO"); // MENSAGEM FIXA
    gotoxy (32,12); // POSICIONAR COLUNA E LINHA
    printf ("[3] EDITAR FUNCIONARIO"); // MENSAGEM FIXA
    gotoxy (32,13); // POSICIONAR COLUNA E LINHA
    printf ("[4] EXCLUIR FUNCIONARIO"); // MENSAGEM FIXA
    gotoxy (32,14); // POSICIONAR COLUNA E LINHA
    printf ("[5] VOLTAR AO MENU ANTERIOR"); // MENSAGEM FIXA
    gotoxy (32,15); // POSICIONAR COLUNA E LINHA
    printf ("[0] SAIR DO SISTEMA"); // MENSAGEM FIXA
    gotoxy (53,6); // POSICIONAR COLUNA E LINHA
    mudacor(PRETO,PRETO); // MUDAR COR DA LINHA
    opcao2 = getche(); // LER COMO CARACTERE E CONVERTE PARA
INTEIRO
    switch (opcao2) // ESCOLHA DE DECISAO
    {
    case '1': // SE APERTAR A OPCAO 1
        incluirfuncionarios(); // DIRECIONANDO PARA CADASTRAR CLIENTE
        fflush(stdin); // LIMPAR O BUFFEN DO TECLADO

```

```

        break; // PARAR
    case '2': // SE APERTAR A OPCAO 2
        consultaFuncionario(); // DIRECIONANDO PARA CONSULTAR CLIENTE
        fflush(stdin); // LIMPAR O BUFFEN DO TECLADO
        break; // PARAR
    case '3': // SE APERTAR A OPCAO 7
        editarfuncionario(); // DIRECIONANDO PARA MENU PRINCIPAL
        fflush(stdin); // LIMPAR O BUFFEN DO TECLADO
        break; // PARAR
    case '4': // SE APERTAR A OPCAO 7
        excluirfuncionario(); // DIRECIONANDO PARA MENU PRINCIPAL
        fflush(stdin); // LIMPAR O BUFFEN DO TECLADO
        break; // PARAR
    case '5': // SE APERTAR A OPCAO 7
        menuGerente(); // DIRECIONANDO PARA MENU PRINCIPAL
        fflush(stdin); // LIMPAR O BUFFEN DO TECLADO
        break; // PARAR
        return;
    default: // SE NENHUMA DAS OPÇÕES ACIMA
        fflush(stdin); // LIMPAR BUFFER TECLADO
        gotoxy (31,20); // POSICIONAR COLUNA E LINHA
        mudacor(BRANCO,VERMELHO); // MUDAR COR DA LINHA
        printf("OPCAO INVALIDA!!!\n\n"); // MENSAGEM FIXA
        sleep (1); // LOOPING DE 1 SEGUNDO
        mudacor(BRANCO,PRETO); // MUDAR COR DA LINHA
        getchar();
        menuFuncionario(); // DIRECIONAMENTO PARA MENU FUNCIONARIO
    }
}

void menuServicos()
{
    system("color 0F"); // ALTERAR A COR DA TELA PARA AZUL E BRANCO
    system("cls"); // LIMPAR TELA
    bordas (); // APLICANDO BORDAS NO MENU CLIENTE
    char opcao2; // VARIABEL
    mudacor(BRANCO,AZUL); // MUDAR COR DA LINHA
    gotoxy (32,3); // POSICIONAR COLUNA E LINHA
    printf (" MENU SERVICOS "); // MENSAGEM FIXA

```

```

mudacor(BRANCO,AZUL); // MUDAR COR DA LINHA
gotoxy (29,6); // POSICIONAR COLUNA E LINHA
printf ("ESCOLHA A OPCAO ABAIXO:"); // MENSAGEM FIXA
mudacor(BRANCO,PRETO); // MUDAR COR DA LINHA
gotoxy (29,10); // POSICIONAR COLUNA E LINHA
printf ("[1] CADASTRAR ORDEM DE SERVICOS"); // MENSAGEM FIXA
gotoxy (29,11); // POSICIONAR COLUNA E LINHA
printf ("[2] LOCALIZAR ORDEM DE SERVICOS"); // MENSAGEM FIXA
gotoxy (29,12); // POSICIONAR COLUNA E LINHA
printf ("[3] CADASTRAR ORCAMENTO CLIENTE"); // MENSAGEM FIXA
gotoxy (29,13); // POSICIONAR COLUNA E LINHA
printf ("[4] LOCALIZAR ORCAMENTO CLIENTE"); // MENSAGEM FIXA
gotoxy (29,14); // POSICIONAR COLUNA E LINHA
printf ("[5] CONSULTAR SERVICOS E VALORES");// MENSAGEM FIXA
gotoxy (29,16); // POSICIONAR COLUNA E LINHA
printf ("[6] VOLTAR AO MENU ANTERIOR"); // MENSAGEM FIXA
gotoxy (29,18); // POSICIONAR COLUNA E LINHA
printf ("[0] SAIR"); // MENSAGEM FIXA
gotoxy (53,6); // POSICIONAR COLUNA E LINHA
mudacor(PRETO,PRETO); // MUDAR COR DA LINHA
    opcao2 = getche(); // LER COMO CARACTERE E CONVERTE PARA
INTEIRO
switch (opcao2) // ESCOLHA DE DECISAO
{
case '1': // SE APERTAR A OPCAO 1
    Cadastrar_OS(); // DIRECIONANDO PARA CADASTRAR CLIENTE
    fflush(stdin); // LIMPAR O BUFFEN DO TECLADO
    break; // PARAR
case '2': // SE APERTAR A OPCAO 2
    consultaOS(); // DIRECIONANDO PARA CONSULTAR CLIENTE
    fflush(stdin); // LIMPAR O BUFFEN DO TECLADO
    break; // PARAR
case '3': // SE APERTAR A OPCAO 3
    incluir_Orcamento(); // DIRECIONANDO PARA CADASTRAR ORDEM
DE SERVICOS
    fflush(stdin); // LIMPAR O BUFFEN DO TECLADO
    break; // PARAR
case '4': // SE APERTAR A OPCAO 4
    consultaOrcamento(); // DIRECIONANDO PARA CONSULTAR ORDEM

```

## DE SERVICOS

```

        fflush(stdin); // LIMPAR O BUFFEN DO TECLADO
        break; // PARAR
    case '5': // SE APERTAR A OPCAO 4
        catalogoServico(); // DIRECIONANDO PARA CONSULTAR ORDEM DE
SERVICOS
        fflush(stdin); // LIMPAR O BUFFEN DO TECLADO
        break;
    case '6': // SE APERTAR A OPCAO 0
        menuGeral(); // DIRECIONANDO PARA SAIR DO SISTEMA
        fflush(stdin); // LIMPAR O BUFFEN DO TECLADO
        break; // PARAR
    case '0': // SE APERTAR A OPCAO 0
        sair(); // DIRECIONANDO PARA SAIR DO SISTEMA
        fflush(stdin); // LIMPAR O BUFFEN DO TECLADO
        break; // PARAR
    default: // SE NENHUMA OPCAO DIGITADA
        fflush(stdin); // LIMPAR O BUFFER DO TECLADO
        gotoxy (31,20); // POSICIONAR COLUNA E LINHA
        mudacor(BRANCO,VERMELHO); // MUDAR COR DA LINHA
        printf("OPCAO INVALIDA!!!\n\n"); // MENSAGEM FIXA
        sleep (1); // LOOPING 1 SEGUNDO
        mudacor(BRANCO,PRETO); // MUDAR COR DA LINHA
        getchar();
        menuServicos(); // DIRECIONAMENTO PARA O MENU SERVICOS
    }
}

void menuGerente()
{
    system("color 0F"); // ALTERAR A COR DA TELA PARA AZUL E BRANCO
    system ("cls"); // LIMPAR TELA
    bordas (); // APLICANDO BORDAS NO MENU GERENTE
    char opcao; // VARIABEL
    gotoxy (33,3); // POSICIONAR COLUNA E LINHA
    mudacor(BRANCO,AZUL); // MUDAR COR DA LINHA
    printf (" MENU GERENTE "); // TITULO
    mudacor(BRANCO,AZUL); // MUDAR COR DA LINHA
    gotoxy (29,6); // POSICIONAR COLUNA E LINHA
    printf ("ESCOLHA A OPCAO ABAIXO:"); // MENSAGEM FIXA

```



```

mudacor(BRANCO,PRETO); // MUDAR COR DA LINHA
gotoxy (30,9); // POSICIONAR COLUNA E LINHA
printf("[1] MENU FUNCIONARIO"); // MENSAGEM FIXA
gotoxy (30,10); // POSICIONAR COLUNA E LINHA
printf("[2] MENU FINANCEIRO"); // MENSAGEM FIXA
gotoxy (30,11); // POSICIONAR COLUNA E LINHA
printf("[3] VOLTAR AO MENU ANTERIOR"); // MENSAGEM FIXA
gotoxy (30,13); // POSICIONAR COLUNA E LINHA
printf("[0] SAIR DO SISTEMA"); // MENSAGEM FIXA
gotoxy (53,6); // POSICIONAR COLUNA E LINHA
mudacor(PRETO,PRETO); //MUDAR COR DA LINHA
    opcao = getche(); // LER COMO CARACTERE E CONVERTE PARA
INTEIRO
    switch (opcao) // ESCOLHA DE DECISAO
    {
        case '1': // SE APERTAR A OPCAO 1
            menuFuncionario(); // DIRECIONANDO PARA CADASTRO DE
FUNCIONARIOS
            fflush(stdin); // LIMPAR O BUFFEN DO TECLADO
            break; // PARAR
        case '2': // SE APERTAR A OPCAO 3
            menuFinanceiro(); // DIRECIONANDO PARA RELATORIO FINANCEIRO
            fflush(stdin); // LIMPAR O BUFFEN DO TECLADO
            break; // PARAR
        case '3': // SE APERTAR A OPCAO 4
            menuGeral(); // DIRECIONANDO PARA VOLTAR AO MENU PRINCIPAL
            fflush(stdin); // LIMPAR O BUFFEN DO TECLADO
            break; // PARAR
        case '0': // SE APERTAR A OPCAO 0
            sair(); // DIRECIONANDO PARA SAIR DO SISTEMA
            fflush(stdin); // LIMPAR O BUFFEN DO TECLADO
            break; // PARAR;
        default: // SE NENHUMA DA OP
            fflush(stdin); // LIMPAR O BUFFER DO TECLADO
            gotoxy (31,20); // POSICIONAMENTO COLUNA E LINHA
            mudacor(BRANCO,VERMELHO); // MUDAR COR DA LINHA
            printf("OPCAO INVALIDA!!!"); // MENSAGEM FIXA
            sleep (1); // LOOPING DE 1 SEGUNDO
            mudacor(BRANCO,PRETO); // MUDAR COR DA LINHA

```

```

        getchar();
        menuGerente(); // DIRECIONAMENTO PARA O MENU GERENTE
    }
}

void menuFinanceiro()
{
    system("color 0F"); // ALTERAR A COR DA TELA PARA AZUL E BRANCO
    system("cls"); // LIMPAR TELA
    bordas (); // APLICANDO BORDAS NO MENU GERENTE
    char opcao; // VARIÁVEL CARACTERE
    gotoxy (32,3); // POSICIONAR COLUNA E LINHA
    mudacor(BRANCO,AZUL); // MUDAR COR DA LINHA
    printf (" MENU FINANCEIRO "); // TÍTULO
    mudacor(BRANCO,AZUL); // MUDAR COR DA LINHA
    gotoxy (29,6); // POSICIONAR COLUNA E LINHA
    printf ("ESCOLHA A OPCAO ABAIXO:"); // MENSAGEM FIXA
    mudacor(BRANCO,PRETO); // MUDAR COR DA LINHA
    gotoxy (30,10);
    printf ("[1] DESPESAS MENSAIS"); // MENSAGEM FIXA
    gotoxy (30,11); // POSICIONAR COLUNA E LINHA
    printf ("[2] GANHOS MENSAIS"); // MENSAGEM FIXA
    gotoxy (30,12); // POSICIONAR COLUNA E LINHA
    printf ("[3] RELATORIO FINANCEIRO"); // MENSAGEM FIXA
    gotoxy (30,13); // POSICIONAR COLUNA E LINHA
    printf ("[4] VOLTAR AO MENU ANTERIOR"); // MENSAGEM FIXA
    gotoxy (30,15); // POSICIONAR COLUNA E LINHA
    printf ("[0] SAIR DO SISTEMA"); // MENSAGEM FIXA
    gotoxy (53,6); // POSICIONAR COLUNA E LINHA
    opcao = getche(); // LER COMO CARACTERE E CONVERTE PARA
INTEIRO
    switch (opcao) // ESCOLHA DE DECISAO
    {
    case '1': // SE APERTAR A OPCAO 1
        despesas(); // DIRECIONANDO PARA CADASTRO DE FUNCIONARIOS
        fflush(stdin); // LIMPAR O BUFFEN DO TECLADO
        break; // PARAR
    case '2': // SE APERTAR A OPCAO 2
        Receita(); // DIRECIONANDO PARA CONSULTA DE FUNCIONARIOS
        fflush(stdin); // LIMPAR O BUFFEN DO TECLADO

```

```

        break; // PARAR;
    case '3': // SE APERTAR A OPCAO 4
        relatorio_Financeiro(); // DIRECIONANDO PARA VOLTAR AO MENU
PRINCIPAL
        fflush(stdin); // LIMPAR O BUFFEN DO TECLADO
        break; // PARAR
    case '4': // SE APERTAR A OPCAO 0
        menuGerente(); // DIRECIONANDO PARA SAIR DO SISTEMA
        fflush(stdin); // LIMPAR O BUFFEN DO TECLADO
        break; // PARAR;
    case '0': // SE APERTAR A OPCAO 0
        sair(); // DIRECIONANDO PARA SAIR DO SISTEMA
        fflush(stdin); // LIMPAR O BUFFEN DO TECLADO
        break; // PARAR;
    default: // SE NENHUMA OPCAO DIGITADA
        fflush(stdin); // LIMPAR O BUFFEN DO TECLADO
        gotoxy (31,20); // POSICIONAMENTO COLUNA E LINHA
        mudacor(BRANCO,VERMELHO); // MUDAR COR DA LINHA
        printf("OPCAO INVALIDA!!!"); // MENSAGEM FIXA
        sleep (1); // LOOPING DE 1 SEGUNDO
        mudacor(BRANCO,PRETO); // MUDAR COR DA LINHA
        getchar();
        menuFinanceiro(); // DIRECIONAMENTO PARA O MENU FINANCEIRO
    }
}
void catalogoServico()
{
    system ("cls"); // LIMPAR TELA
    mudacor(BRANCO,PRETO);
    bordas(); // APLICANDO BORDAS NA TELA CONSULTA
    gotoxy (25,3); // POSICIONAR COLUNA E LINHA
    mudacor(BRANCO,AZUL); // MUDAR COR DA LINHA
    printf ("AREA DE CONSULTA PRECOS E SERVICOS ");// MENSAGEM
FIXA
    mudacor(BRANCO,PRETO); // MUDAR COR DA LINHA
    gotoxy (31,5); // POSICIONAR COLUNA E LINHA
    printf(" Valores de Mao de Obra Tabelados");// MENSAGEM FIXA
    gotoxy(30,6); // POSICIONAR COLUNA E LINHA
    printf(" Os valores nao inclui Pecas!!!");// MENSAGEM FIXA

```

```

gotoxy(4,8); // POSICIONAR COLUNA E LINHA
printf("Formatacao: R$ 250,00"); // MENSAGEM FIXA
gotoxy(4,9); // POSICIONAR COLUNA E LINHA
printf("Reparo na Placa Mae: R$ 850,00");// MENSAGEM FIXA
gotoxy(4,10); // POSICIONAR COLUNA E LINHA
printf("Troca da Tela: R$ 420,00"); // MENSAGEM FIXA
gotoxy(4,11); // POSICIONAR COLUNA E LINHA
printf("Troca do Teclado: R$ 350,00"); // MENSAGEM FIXA
gotoxy(4,12); // POSICIONAR COLUNA E LINHA
printf("Backup: R$ 280,00"); // MENSAGEM FIXA
gotoxy(4,13); // POSICIONAR COLUNA E LINHA
printf("Deslocamento por hora: R$ 150,00");// MENSAGEM FIXA
gotoxy(10,22); // POSICIONAR COLUNA E LINHA
mudacor(PRETO,PRETO); // MUDAR COR DA LINHA
system ("pause"); // PAUSAR MENSAGEM NA TELA
mudacor(BRANCO,PRETO); // MUDAR COR DA LINHA
menuGeral(); // DIRECIONAMENTO PARA O MENU GERAL
}
void despesas()
{
    struct Despesas;
    char R_Financeiro[1000];
    float Conta_agua;
    float Conta_luz;
    float Conta_internet;
    float Conta_aluguel;
    float pgto_funcionario;
    float Conta_extras;
    float total_despesas;
    char data_despesa[50];
    int soma;
    system("cls"); // LIMPAR TELA
    mudacor(BRANCO,PRETO);
    bordas(); // APLICANDO BORDAS
    gotoxy (31,3); // POSICIONAMENTO COLUNA E LINHA
    mudacor(BRANCO,AZUL); // MUADR COR DA LINHA
    printf (" DESPESAS MENSAIS "); // MENSAGEM FIXA
    mudacor(BRANCO,PRETO); // MUDAR COR DA LINHA
    FILE *Financeiro_arquivo; // CRIANDO PONTEIRO

```

```

Financeiro_arquivo = fopen("financeiro.txt","ab+"); // CRIANDO ARQUIVO
gotoxy(6,7);
printf("MES/ANO.....");
gotoxy(40,7);
scanf("%s",&data_despesa);
fflush(stdin);
gotoxy(6,8); // POSICIONAMENTO COLUNA E LINHA
printf("CONTA DE AGUA R$....."); // MENSAGEM FIXA
gotoxy(40,8); // POSICIONAMENTO COLUNA E LINHA
scanf("%f",&Conta_agua); // DIGITAR O DADO
fflush(stdin); // LIMPAR BUFFER DO TECLADO
gotoxy(6,9); // POSICIONAMENTO COLUNA E LINHA
printf("CONTA DE LUZ R$....."); // MENSAGEM FIXA
gotoxy(40,9); // POSICIONAMENTO COLUNA E LINHA
scanf("%f",&Conta_luz); // DIGITAR O DADO
fflush(stdin); // LIMPAR BUFFER DO TECLADO
gotoxy(6,10); // POSICIONAMENTO COLUNA E LINHA
printf("CONTA DE INTERNET R$....."); // MENSAGEM FIXA
gotoxy(40,10); // POSICIONAMENTO COLUNA E LINHA
scanf("%f",&Conta_internet); // DIGITAR O DADO
fflush(stdin); // LIMPAR BUFFER DO TECLADO
gotoxy(6,11); // POSICIONAMENTO COLUNA E LINHA
printf("ALUGUEL R$....."); // MENSAGEM FIXA
gotoxy(40,11); // POSICIONAMENTO COLUNA E LINHA
scanf("%f",&Conta_aluguel); // DIGITAR O DADO
fflush(stdin); // LIMPAR BUFFER DO TECLADO
gotoxy(6,12); // POSICIONAMENTO COLUNA E LINHA
printf("PAGAMENTO DE FUNCIONARIOS R$."); // MENSAGEM FIXA
gotoxy(40,12); // POSICIONAMENTO COLUNA E LINHA
scanf("%f",&pgto_funcionario); // DIGITAR O DADO
fflush(stdin); // LIMPAR BUFFER DO TECLADO
gotoxy(6,13); // POSICIONAMENTO COLUNA E LINHA
printf("DESPESAS EXTRAS R$....."); // MENSAGEM FIXA
gotoxy(40,13); // POSICIONAMENTO COLUNA E LINHA
scanf("%f",&Conta_extras); // DIGITAR O DADO
fflush(stdin); // LIMPAR BUFFER DO TECLADO
gotoxy(6,15); // POSICIONAMENTO COLUNA E LINHA
mudacor(BRANCO,VERMELHO);

```

```

        printf("TOTAL BRUTO DE DESPESAS R$...: " " %.2f",
(Conta_agua+Conta_luz+Conta_internet+Conta_aluguel+pgto_funcionario+Conta_e
xtras)); // MENSAGEM FIXA
        mudacor(BRANCO,PRETO);
        if (Financeiro_arquivo !=NULL) // SE O ARQUIVO EXISTIR, GRAVAR
        {
            fprintf(Financeiro_arquivo,"DESPESAS - CTECH TECNOLOGIA\n"); //
TITUTO
                fprintf(Financeiro_arquivo,"MES/ANO.....:
""%s\n",data_despesa); //SALVANDO NO ARQUIVO
                fprintf(Financeiro_arquivo,"CONTA DE AGUA.....R$:
""%.2fn",Conta_agua); // SALVANDO NO ARQUIVO
                fprintf(Financeiro_arquivo,"CONTA DE LUZ .....R$:
""%.2fn",Conta_luz); // SALVANDO NO ARQUIVO
                fprintf(Financeiro_arquivo,"CONTA DE INTERNET.....R$:
""%.2fn",Conta_internet);// SALVANDO NO ARQUIVO
                fprintf(Financeiro_arquivo,"ALUGUEL .....R$:
""%.2fn",Conta_aluguel); // SALVANDO NO ARQUIVO
                fprintf(Financeiro_arquivo,"PAGAMENTO DE FUNCIONARIOS.... R$:
""%.2fn",pgto_funcionario);// SALVANDO NO ARQUIVO
                fprintf(Financeiro_arquivo,"DESPESAS EXTRAS..... R$:
""%.2fn",Conta_extras); // SALVANDO NO ARQUIVO
                fprintf(Financeiro_arquivo,"TOTAL BRUTO DE DESPESAS.....R$:
""%.2fn\n",
(Conta_agua+Conta_luz+Conta_internet+Conta_aluguel+pgto_funcionario+Conta_e
xtras)); // SALVANDO NO ARQUIVO
                fclose(Financeiro_arquivo); // FECHANDO O PONTEIRO
                gotoxy(4,20); // POSICIONAMENTO COLUNA E LINHA
                mudacor(BRANCO,VERDE); // MUDAR COR DA LINHA
                gotoxy (20,20); // POSICIONAMENTO COLUNA E LINHA
                printf("DESPESAS REGISTRADAS COM SUCESSO!!!");//
MENSAGEM FIXA
                getchar();
                mudacor(BRANCO,PRETO);
                menuFinanceiro(); // DIRECIONAMENTO PARA O MENU GERENTE
        }
        else // SE NAO EXISTIR
        {

```

```

        mudacor(BRANCO,VERMELHO); // ALTERAR COR DA LINHA
        gotoxy (10,24); // POSICIONAMENTO COLUNA E LINHA
        printf("ERRO AO CADASTRAR FINANÇAS, TENTE NOVAMENTE!!!"); //
MENSAGEM FIXA
        mudacor(BRANCO,PRETO); // ALTERAR COR DA LINHA
        menuGerente();
    }
}
void Receita()
{
    struct Receita;
    char R_Financeiro[1000];
    float servicos;
    float outros;
    char data_receita[50];
    system("cls"); // LIMPA TELA
    mudacor(BRANCO,PRETO);
    bordas(); // MARGENS
    gotoxy (31,3); // POSICIONAMENTO DE COLUNAS E LINHAS
    mudacor(BRANCO, AZUL); // MUDAR COR DA LINHA
    printf (" RECEITAS MENSAS "); // MENSAGEM FIXA
    gotoxy (31,5); // POSICIONAMENTO DE COLUNAS E LINHAS
    mudacor(BRANCO, PRETO); // MUDAR COR DA LINHA
    FILE *Financeiro_arquivo; // CRIAR PONTEIRO
    Financeiro_arquivo = fopen("financeiro.txt","ab+"); // CRIANDO ARQUIVO
    gotoxy(6,7);
    printf("MES/ANO.....:");
    gotoxy(37,7);
    scanf("%s",&data_receita);
    fflush(stdin);
    gotoxy(6,8); // POSICIONAMENTO DE COLUNAS E LINHAS
    printf("SERVICOS R$.....:"); // MENSAGEM FIXA
    gotoxy(37,8); // POSICIONAMENTO DE COLUNAS E LINHAS
    scanf("%f",&servicos); // LEITURA DE DADOS
    fflush(stdin); // LIMPAR BUFFER DO TECLADO
    gotoxy(6,9); // POSICIONAMENTO DE COLUNAS E LINHAS
    printf("OUTROS R$.....:"); // MENSAGEM FIXA
    gotoxy(37,9); // POSICIONAMENTO DE COLUNAS E LINHAS

```

```

scanf("%f",&outros); // LEITURA DE DADOS
fflush(stdin); // LIMPAR BUFFER DO TECLADO
gotoxy(6,11); // POSICIONAMENTO DE COLUNAS E LINHAS
printf("TOTAL DE RECEITAS R$.....:" " %.2f",(servicos+outros)); //
MENSAGEM FIXA
if (Financeiro_arquivo!=NULL) // ABRINDO ARQUIVO
{
    fprintf(Financeiro_arquivo,"RECEITA - CTECH TECNOLOGIA\n");
//GRAVANDO NO ARQUIVO
    fprintf(Financeiro_arquivo,"MES/ANO.....:
""%s\n",data_receita);
    fprintf(Financeiro_arquivo,"SERVICOS.....R$:
""%.2f\n",servicos); //GRAVANDO NO ARQUIVO
    fprintf(Financeiro_arquivo,"OUTROS .....R$: ""%.2f\n",outros);
//GRAVANDO NO ARQUIVO
    fprintf(Financeiro_arquivo,"TOTAL DE RECEITAS.....R$: ""%.2f\n\n",
(servicos+outros)); //GRAVANDO NO ARQUIVO
    fclose(Financeiro_arquivo); // FECHANDO E SALVANDO ARQUIVO
    gotoxy(4,20); // POSICIONAMENTO DE COLUNAS E LINHAS
    mudacor(BRANCO,VERDE); // MUDAR COR DA LINHA
    gotoxy (20,20); // POSICIONAMENTO DE COLUNAS E LINHAS
    printf("RECEITAS REGISTRADAS COM SUCESSO!!!"); //
MENSAGEM FIXA
    getchar();
    mudacor(BRANCO,PRETO);
    menuFinanceiro();
}
while (fgets(R_Financeiro,1000,Financeiro_arquivo)!= NULL)
{
    mudacor(BRANCO,VERDE);
    printf("RECEITAS MENSAIS: %f");
    mudacor(BRANCO,PRETO);
    gotoxy (6,8);
    printf("SERVICOS.....: ""%.2f",servicos);
    gotoxy (6,9);
    printf("OUTRAS RECEITAS.....: ""%.2f",outros);
    gotoxy (6,11);
    printf("TOTAL DE RECEITAS.....: ""%f",(servicos+outros));

```



```

    }
    fclose(Financeiro_arquivo);
    gotoxy (4,20);
    getch();
    menuGerente();
}
void relatorio_Financeiro()
{
    system("cls"); // limpeza de tela
    mudacor(BRANCO,PRETO);
    bordas();
    gotoxy (31,3);
    mudacor(BRANCO, AZUL);
    printf (" RELATORIO FINANCEIRO ");
    gotoxy (31,5);
    mudacor(BRANCO, PRETO);
    float total_receita;
    float total_despesas;
    mudacor(BRANCO,VERDE);
    gotoxy(10,8);
    printf("TOTAL DE RECEITAS MENSAL R$: ");
    mudacor(BRANCO,PRETO);
    gotoxy(40,8);
    scanf("%f",&total_receita);
    mudacor(BRANCO,PRETO);
    mudacor(BRANCO,VERMELHO);
    gotoxy (10,12);
    printf("TOTAL DE DESPESAS MENSAL R$: ");
    gotoxy (40,12);
    mudacor(BRANCO,PRETO);
    scanf("%f",&total_despesas);
    mudacor(PRETO,AMARELO);
    gotoxy(31,17);
    printf("LUCRO DA EMPRESA");
    mudacor(BRANCO,PRETO);
    gotoxy(34,18);
    printf("R$ %.2f",(total_receita-total_despesas));
    mudacor(BRANCO,PRETO);

```

```

gotoxy(15,20);
mudacor(PRETO,PRETO);
system("pause");
mudacor(BRANCO,PRETO);
getchar();
menuFinanceiro();
}
void Cadastrar_OS()
{
    struct Cadastrar_OS;
    char C_OS[1000];
    int codigoOS;
    char Marca_OS[10];
    char Modelo_OS [100];
    char Numero_de_serie_OS [100];
    char Defeito_Relatado_OS [100];
    char CPF_CNPJ_OS[100];
    char Nome_Completo_OS[100];
    char Data_OS [100];
    char Telefone_OS[100];
    char Estado_OS[100];
    char Cidade_OS[100];
    char hora_OS[100];
    char Servicos_Executado_OS[100];
    char Tecnico_Responsavel_OS[100];
    float valor_OS;
    system("cls"); // limpeza de tela
    mudacor(BRANCO,PRETO);
    bordas();
    mudacor(BRANCO,AZUL);
    gotoxy (31,3);
    printf (" CADASTRAR ORDEM DE SERVICO ");
    mudacor(BRANCO,PRETO);
    FILE *OS_arquivo;
    OS_arquivo = fopen("C_OS.txt","ab+");
    if (OS_arquivo ==NULL)
    {
        gotoxy(10,10);
        mudacor(BRANCO,VERMELHO);
    }
}

```

```

printf("NAO FOI POSSIVEL ABRIR O ARQUIVO");
mudacor(BRANCO,PRETO);
getchar();
menuGeral();
}
else
{
    gotoxy(6,6);
    printf("CODIGO.....: ");
    gotoxy(28,6);
    scanf("%d",&codigoOS);
    fflush(stdin);
    fprintf(OS_arquivo,"\nCODIGO.....: ""%d",codigoOS);
    gotoxy(6,8);
    printf("DATA DA OS.....:");
    gotoxy(28,8);
    scanf("%s",&Data_OS);
    fflush(stdin);
    fprintf(OS_arquivo,"\nDATA DA OS.....: ""%s",&Data_OS);
    gotoxy(6,9);
    printf("NOME DO CLIENTE.....:");
    gotoxy(28,9);
    fgets(Nome_Completo_OS,100,stdin);
    Nome_Completo_OS[strlen(Nome_Completo_OS)-1] = '\0';
    fprintf(OS_arquivo,"\nNOME DO CLIENTE.....: "
"%s",&Nome_Completo_OS);
    gotoxy(6,10);
    printf("CPF/CNPJ.....:");
    gotoxy(28,10);
    scanf("%s",&CPF_CNPJ_OS);
    fflush(stdin);
    fprintf(OS_arquivo,"\nCPF/CNPJ.....: " "%s",&CPF_CNPJ_OS);
    gotoxy(6,11);
    printf("TELEFONE.....:");
    gotoxy(28,11);
    scanf("%s",&Telefone_OS);
    fflush(stdin);
    fprintf(OS_arquivo,"\nTELEFONE.....: " "%s",&Telefone_OS);
    gotoxy(6,12);

```

```

printf("MARCA.....:");
gotoxy(28,12);
scanf("%s",&Marca_OS);
fflush(stdin);
fprintf(OS_arquivo,"\nMARCA.....: " "%s",&Marca_OS);
gotoxy(6,13);
printf("MODELO.....:");
gotoxy(28,13);
scanf("%s",&Modelo_OS);
fflush(stdin);
fprintf(OS_arquivo,"\nMODELO.....: " "%s",&Modelo_OS);
gotoxy(6,14);
printf("SERIE.....:");
gotoxy(28,14);
scanf("%s",&Numero_deSerie_OS);
fflush(stdin);
                                fprintf(OS_arquivo,"\nSERIE.....: "
"%s",&Numero_deSerie_OS);
gotoxy(6,15);
printf("PROBLEMA RELATADO....:");
gotoxy(28,15);
fgets(Defeito_Relatado_OS,100,stdin);
Defeito_Relatado_OS[strlen(Defeito_Relatado_OS)-1] = '\0';
                                fprintf(OS_arquivo,"\nPROBLEMA  RELATADO.....: "
"%s",&Defeito_Relatado_OS);
gotoxy(6,16);
printf("SERVICOS EXECUTADOS.:");
gotoxy(28,16);
fgets(Servicos_Executado_OS,100,stdin);
Servicos_Executado_OS[strlen(Servicos_Executado_OS)-1] = '\0';
                                fprintf(OS_arquivo,"\nSERVICOS  EXECUTADOS.....: "
"%s",&Servicos_Executado_OS);
gotoxy(6,17);
printf("TECNICO RESPONSABEL.:");
gotoxy(28,17);
fgets(Tecnico_Responsavel_OS,100,stdin);
Tecnico_Responsavel_OS[strlen(Tecnico_Responsavel_OS)-1] = '\0';
                                fprintf(OS_arquivo,"\nTECNICO  RESPONSABEL.....:
""%s",&Tecnico_Responsavel_OS);

```

```

        gotoxy(6,19);
        printf("VALOR DO SERVICO REALIZADOS R$:");
        gotoxy(39,19);
        scanf("%f",&valor_OS);
        fflush(stdin);
        fprintf(OS_arquivo,"\nVALOR DO SERVICO REALIZADOS R$: "
"%0.2f",valor_OS);
        fclose(OS_arquivo);
        gotoxy (15,22);
        mudacor(BRANCO,VERDE);
        printf("ORDEM DE SERVICO REGISTRADOS COM SUCESSO!!!");
        sleep (1);
        mudacor(BRANCO,PRETO);
        menuGeral();
    }
}

void consultaOS()
{
    system("cls"); // limpeza de tela
    mudacor(BRANCO,PRETO);
    bordas();
    gotoxy (31,3);
    mudacor(BRANCO, AZUL);
    printf (" CONSULTA ORDEM DE SERVICOS ");
    gotoxy (31,5);
    mudacor(BRANCO, PRETO);
    struct Cadastrar_OS;
    char C_OS[1000];
    FILE *OS_arquivo;
    OS_arquivo = fopen("C_OS.txt","r");
    if (OS_arquivo==NULL)
    {
        gotoxy(23,15);
        mudacor(BRANCO,VERMELHO);
        printf("NAO FOI POSSIVEL LOCALIZAR O ARQUIVO.");// caso nao
consiga acessar o arquivo
        mudacor(BRANCO,PRETO);
        getchar();
    }
}

```

```

while (fgets(C_OS,1000,OS_arquivo)!= NULL)
{
    printf("\t %s",&C_OS);
}
fclose(OS_arquivo);
getchar();
mudacor(BRANCO,PRETO);
menuGeral();
}
void incluir_Orcamento()
{
    struct incluir_Orcamento;
    char C_Orcamento[1000];
    int codigoOrcamento;
    char Data_Orcamento [20];
    char Marca_Orcamento[50];
    char Modelo_Orcamento [50];
    char Numero_de_serie_Orcamento [20];
    char Defeito_Relatado_Orcamento [200];
    char Possui_arranhoes_Orcamento [10];
    char Possui_Bateria_Orcamento [10];
    char Possui_fonte_Orcamento [10];
    char nome_do_cliente_Orcamento[100];
    char telefone_Orcamento[20];
    char CPF_CNPJ_Orcamento[50];
    system("cls"); // LIMPAR TELA
    mudacor(BRANCO,PRETO);
    bordas(); //APLICAR A BORDA
    gotoxy (31,3); //POSICIONAMENTO DE LINHAS E COLUNAS
    mudacor(BRANCO,AZUL); // ALTERAR A COR DA LETRA
    printf (" CADASTRO DE ORCAMENTO "); // MENSAGEM FIXA
    mudacor(BRANCO,PRETO);
    FILE *Orcamento_arquivo; // CRIAR O PONTEIRO
    Orcamento_arquivo = fopen("Orcamento.txt","ab+"); // CRIAR O ARQUIVO
    if(Orcamento_arquivo ==NULL) // FAZENDO A LEITURA DO ARQUIVO
    {
        gotoxy(23,15);
        mudacor(BRANCO,VERMELHO);
        printf("NAO FOI LOCALIZAR O ARQUIVO"); // MENSAGEM FIXA
    }
}

```

```

        mudacor(BRANCO,PRETO);
        getchar();
    }
    else // CASO ESTEJA TUDO CERTO, SEGUIR
    {
        gotoxy(6,6); //POSICIONAMENTO DE LINHAS E COLUNAS
        printf("CODIGO.....:"); // MENSAGEM FIXA
        gotoxy(28,6); //POSICIONAMENTO DE LINHAS E COLUNAS
        scanf("%d",&codigoOrcamento); // DIGITAR O CAMPO
        fflush(stdin); // LIMPAR BUFFER DO TECLADO
        fprintf(Orcamento_arquivo,  "\nCODIGO.....:"
"%d",codigoOrcamento); // SALVAR NO ARQUIVO
        gotoxy(6,8); //POSICIONAMENTO DE LINHAS E COLUNAS
        printf("DATA DO ORCAMENTO....:"); // MENSAGEM FIXA
        gotoxy(28,8); //POSICIONAMENTO DE LINHAS E COLUNAS
        scanf("%s",&Data_Orcamento); // DIGITAR O CAMPO
        fflush(stdin); // LIMPAR BUFFER DO TECLADO
        fprintf(Orcamento_arquivo, "\nDATA DO ORCAMENTO....:" "%s",
&Data_Orcamento); // SALVAR NO ARQUIVO
        gotoxy(6,9); //POSICIONAMENTO DE LINHAS E COLUNAS
        printf("NOME DO CLIENTE.....:"); // MENSAGEM FIXA
        gotoxy(28,9); //POSICIONAMENTO DE LINHAS E COLUNAS
        fgets(&nome_do_cliente_Orcamento,100,stdin);// DIGITAR O CAMPO
        nome_do_cliente_Orcamento[strlen(nome_do_cliente_Orcamento)-1] =
'\0'; // LIMPAR QUEBRA DE LINHA
        fflush(stdin);
        fprintf(Orcamento_arquivo,"\nNOME DO CLIENTE.....:" "%s",
&nome_do_cliente_Orcamento); // SALVAR NO ARQUIVO
        gotoxy(6,10); //POSICIONAMENTO DE LINHAS E COLUNAS
        printf("CPF/CNPJ.....:"); // MENSAGEM FIXA
        gotoxy(28,10); //POSICIONAMENTO DE LINHAS E COLUNAS
        scanf("%s",&CPF_CNPJ_Orcamento); // DIGITAR O CAMPO
        fflush(stdin); // LIMPAR BUFFER DO TECLADO
        fprintf(Orcamento_arquivo,  "\nCPF/CNPJ.....:" "%s",
&CPF_CNPJ_Orcamento); // SALVAR NO ARQUIVO
        gotoxy(6,11); //POSICIONAMENTO DE LINHAS E COLUNAS
        printf("TELEFONE.....:"); // MENSAGEM FIXA
        gotoxy(28,11); //POSICIONAMENTO DE LINHAS E COLUNAS

```

```

scanf("%s",&telefone_Orcamento); // DIGITAR O CAMPO
fflush(stdin); // LIMPAR BUFFER DO TECLADO
        fprintf(Orcamento_arquivo, "\nTELEFONE.....:" "%s",
&telefone_Orcamento); // SALVAR NO ARQUIVO
        gotoxy(6,12); //POSICIONAMENTO DE LINHAS E COLUNAS
        printf("MARCA.....:"); // MENSAGEM FIXA
        gotoxy(28,12); //POSICIONAMENTO DE LINHAS E COLUNAS
        fgets(&Marca_Orcamento,100,stdin); // DIGITAR O CAMPO
        Marca_Orcamento[strlen(Marca_Orcamento)-1] = '\0'; // LIMPAR
QUEBRA DE LINHA
        fflush(stdin);
        fprintf(Orcamento_arquivo, "\nMARCA.....:" "%s",
&Marca_Orcamento); // SALVAR NO ARQUIVO
        gotoxy(6,13); //POSICIONAMENTO DE LINHAS E COLUNAS
        printf("MODELO.....:"); // MENSAGEM FIXA
        gotoxy(28,13); //POSICIONAMENTO DE LINHAS E COLUNAS
        fgets(&Modelo_Orcamento,100,stdin); // DIGITAR O CAMPO
        Modelo_Orcamento[strlen(Modelo_Orcamento)-1] = '\0'; // LIMPAR
QUEBRA DE LINHA
        fflush(stdin);
        fprintf(Orcamento_arquivo, "\nMODELO.....:" "%s",
&Modelo_Orcamento); // SALVAR NO ARQUIVO
        gotoxy(6,14); //POSICIONAMENTO DE LINHAS E COLUNAS
        printf("SERIE.....:"); // MENSAGEM FIXA
        gotoxy(28,14); //POSICIONAMENTO DE LINHAS E COLUNAS
        scanf("%s",&Numero_de_serie_Orcamento); // DIGITAR O CAMPO
        fflush(stdin); // LIMPAR BUFFER DO TECLADO
        fprintf(Orcamento_arquivo, "\nSERIE.....:" "%s",
&Numero_de_serie_Orcamento); // SALVAR NO ARQUIVO
        gotoxy(6,16); //POSICIONAMENTO DE LINHAS E COLUNAS
        printf("POSSUI BATERIA....."); // MENSAGEM FIXA
        gotoxy(28,16); //POSICIONAMENTO DE LINHAS E COLUNAS
        scanf("%s",&Possui_Bateria_Orcamento); // DIGITAR O CAMPO
        fflush(stdin); // LIMPAR BUFFER DO TECLADO
        fprintf(Orcamento_arquivo, "\nPOSSUI BATERIA.....:" "%s",
&Possui_Bateria_Orcamento); // SALVAR NO ARQUIVO
        gotoxy(6,18); //POSICIONAMENTO DE LINHAS E COLUNAS
        printf("DEFEITO RELATADO.....:"); // MENSAGEM FIXA

```



```

        gotoxy(28,18); //POSICIONAMENTO DE LINHAS E COLUNAS
        fgets(&Defeito_Relatado_Orcamento,200,stdin); // DIGITAR O CAMPO
        Defeito_Relatado_Orcamento[strlen(Defeito_Relatado_Orcamento)-1] =
'\0'; // LIMPAR QUEBRA DE LINHA
        fflush(stdin);
        fprintf(Orcamento_arquivo, "\nDEFEITO RELATADO....:" "%s",
&Defeito_Relatado_Orcamento); // SALVAR NO ARQUIVO
        fclose(Orcamento_arquivo);
        gotoxy (20,22); //POSICIONAMENTO DE LINHAS E COLUNAS
        mudacor(BRANCO,VERDE); // MUDAR COR DA LETRA
        printf("ORCAMENTO REGISTRADO COM SUCESSO!!!"); //
MENSAGEM FIXA
        mudacor(BRANCO,PRETO);
        sleep (1); // TEMPO PARA PULAR A TELA
        menuGeral(); // DIRECIONAMENTO APOS CADASTRO
    }
}
void consultaOrcamento()
{
    system("cls"); // limpeza de tela
    mudacor(BRANCO,PRETO);
    bordas();
    gotoxy (31,3);
    mudacor(BRANCO,AZUL);
    printf (" CONSULTA ORCAMENTO ");
    gotoxy (31,5);
    mudacor(BRANCO, PRETO);
    struct incluir_Orcamento;
    char C_Orcamento[1000];
    FILE *Orcamento_arquivo;
    Orcamento_arquivo = fopen ("Orcamento.txt","r");
    if (Orcamento_arquivo==NULL)
    {
        gotoxy(23,15);
        mudacor(BRANCO,VERMELHO);
        printf("NAO FOI POSSIVEL ACESSAR O ARQUIVO.");
        mudacor(BRANCO,PRETO);
        getchar();
    }
}

```

```

    }
    while (fgets(C_Orcamento,1000,Orcamento_arquivo)!= NULL)
    {
        printf("\t %s",C_Orcamento);
    }
    fclose(Orcamento_arquivo);
    gotoxy (4,24);
    getchar();
    menuGeral();
}
void incluirCliente()
{
    struct incluir_Cliente;
    char C_Cliente[10];
    char nome_Cliente [100];
    char endereco_Cliente [200];
    char telefone_Cliente [50];
    char estado_Cliente [50];
    char cpf_Cliente [14];
    char cep_Cliente [14];
    int codigo_Cliente;
    char data_nasc_Cliente [20];
    char cidade_Cliente [20];
    system("cls"); // limpeza de tela
    mudacor(BRANCO,PRETO);
    bordas();
    gotoxy (31,3);
    mudacor(BRANCO, AZUL);
    printf (" CADASTRO DE CLIENTE ");
    mudacor(BRANCO,PRETO);
    gotoxy (10,10);
    FILE *Cliente_arquivo;
    Cliente_arquivo = fopen("Cliente.txt","ab+");
    if (Cliente_arquivo ==NULL)
    {
        mudacor(BRANCO,VERMELHO);
        printf("NAO FOI POSSIVEL ABRIR OU CRIAR O ARQUIVO");
        mudacor(BRANCO, PRETO);
    }
}

```

```

        return(0);
        system("cls");
    }
    else
    {
        gotoxy(6,7);
        printf("CODIGO.....:");
        gotoxy(24,7);
        scanf("%d",&codigo_Cliente);
        fflush(stdin);
        fprintf(Cliente_arquivo, "\nCODIGO.....:" "%d",codigo_Cliente);
        gotoxy(6,9);
        printf("NOME COMPLETO.....:");
        fgets(&nome_Cliente,100,stdin);
        nome_Cliente[strlen(nome_Cliente)-1] = '\0';
        fflush(stdin);
        gotoxy(24,9);
        fprintf(Cliente_arquivo, "\nNOME COMPLETO.....:" "%s",&nome_Cliente);
        gotoxy(6,10);
        printf("DATA DE NASC.....:");
        gotoxy(24,10);
        scanf("%s",&data_nasc_Cliente);
        fflush(stdin);
        fprintf(Cliente_arquivo, "\nDATA DE NASC.....:"
"%s",&data_nasc_Cliente);
        gotoxy(6,11);
        printf("CPF/CNPJ.....:");
        gotoxy(24,11);
        scanf("%s",&cpf_Cliente);
        fflush(stdin);
        fprintf(Cliente_arquivo, "\nCPF/CNPJ.....:" "%s", &cpf_Cliente);
        gotoxy(6,12);
        printf("TELEFONE.....:");
        gotoxy(24,12);
        scanf("%s",&telefone_Cliente);
        fflush(stdin);
        fprintf(Cliente_arquivo, "\nTELEFONE.....:" "%s", &telefone_Cliente);
        gotoxy(6,13);

```

```

printf("ENDERECO.....");
gotoxy(24,13);
fgets(&endereco_Cliente,200,stdin);
endereco_Cliente[strlen(endereco_Cliente)-1] = '\0';
fflush(stdin);
fprintf(Cliente_arquivo, "\nENDERECO....." "%s", &endereco_Cliente);
gotoxy(6,14);
printf("CIDADE.....");
gotoxy(24,14);
fgets(&cidade_Cliente,20,stdin);
cidade_Cliente[strlen(cidade_Cliente)-1] = '\0';
fflush(stdin);
fprintf(Cliente_arquivo, "\nCIDADE....." "%s", &cidade_Cliente);
gotoxy(6,15);
printf("ESTADO.....");
gotoxy(24,15);
fgets(&estado_Cliente,50,stdin);
estado_Cliente[strlen(estado_Cliente)-1] = '\0';
fflush(stdin);
fprintf(Cliente_arquivo, "\nESTADO....." "%s", &estado_Cliente);
gotoxy(6,16);
printf("CEP.....");
gotoxy(24,16);
scanf("%s",&cep_Cliente);
fflush(stdin);
fprintf(Cliente_arquivo, "\nCEP....." "%s", &cep_Cliente);
}
gotoxy (20,18);
mudacor(BRANCO,VERDE);
printf("CLIENTE CADASTRADO COM SUCESSO!!!");
fclose(Cliente_arquivo);
mudacor(BRANCO,PRETO);
gotoxy (15,25);
sleep (1);
menuGeral();
}
void editarCliente()
{
    system("cls"); // limpeza de tela

```

```

    mudacor(BRANCO,PRETO);
    bordas();
    gotoxy (33,3);
    mudacor(BRANCO, AZUL);
    printf (" EDITAR CLIENTE ");
    mudacor(BRANCO,PRETO);
    mudacor(PRETO,AMARELO);
    gotoxy (28,10);
    printf("AREA EM DESENVOLVIMENTO!!! ");
    gotoxy (25,12);
    printf("EM BREVE ATUALIZACAO DO SOFTWARE");
    mudacor(BRANCO,PRETO);
    gotoxy(20,24);
    getch();
    menuGeral();
}

void excluirCliente()
{
    system("cls"); // limpeza de tela
    mudacor(BRANCO,PRETO);
    bordas();
    gotoxy (33,3);
    mudacor(BRANCO, AZUL);
    printf (" EXCLUIR CLIENTE ");
    mudacor(BRANCO,PRETO);
    mudacor(PRETO,AMARELO);
    gotoxy (28,10);
    printf("AREA EM DESENVOLVIMENTO!!! ");
    gotoxy (25,12);
    printf("EM BREVE ATUALIZACAO DO SOFTWARE");
    mudacor(BRANCO,PRETO);
    gotoxy(20,24);
    getch();
    menuGeral();
}

void consultaCliente()
{
    system("cls"); // limpeza de tela
    mudacor(BRANCO,PRETO);

```

```

bordas();
gotoxy (31,3);
mudacor(BRANCO,AZUL);
printf (" CONSULTA CLIENTE ");
gotoxy (31,5);
mudacor(BRANCO, PRETO);
struct incluir_Cliente;
char C_Cliente[100000];
FILE *Cliente_arquivo;
Cliente_arquivo = fopen("Cliente.txt", "r");
if (Cliente_arquivo == NULL)
{
    gotoxy(23,15);
    mudacor(BRANCO,VERMELHO);
    printf("NAO FOI POSSIVEL LOCALIZAR O ARQUIVO!!!");
    mudacor(BRANCO,PRETO);
    getchar();
}
while (fgets(C_Cliente,1000,Cliente_arquivo)!= NULL)
{
    printf("\t %s",C_Cliente);
}
fclose(Cliente_arquivo);
gotoxy(6,30);
getchar();
menuCliente();
}
void incluirfuncionarios()
{
    struct incluir_Funcionarios;
    int C_Funcionarios[SIZE];
    char nome_Funcionario[SIZE] [100];
    char endereco_Funcionario[SIZE] [200];
    char telefone_Funcionario[SIZE] [50];
    char rg_Funcionario[SIZE][12];
    char cpf_Funcionario [SIZE][14];
    char funcao_Funcionario[SIZE][50];
    char data_admissao_Funcionario[SIZE][20];
    char data_nasc_Funcionario[SIZE][20];

```

```

char estado_Funcionario[SIZE][50];
char cidade_Funcionario[SIZE][20];
char cep_Funcionario[SIZE][10];
system("cls"); // limpeza de tela
mudacor(BRANCO,PRETO);
bordas(); // APLICANDO BORDAS
gotoxy (31,3);
mudacor(BRANCO,AZUL);
printf (" CADASTRO DE FUNCIONARIOS ");
mudacor(BRANCO,PRETO);
FILE *Funcionario_arquivo;
Funcionario_arquivo = fopen("Funcionario.txt","ab+");
if (Funcionario_arquivo ==NULL)
{
    printf("NAO FOI POSSIVEL ABRIR OU CRIAR O ARQUIVO");
    system("pause");
}
else
{
    gotoxy(6,9);
    printf("NOME COMPLETO.....");
    gotoxy(27,9);
    fgets(nome_Funcionario[SIZE],100,stdin);
    nome_Funcionario[SIZE][strlen(nome_Funcionario[SIZE])-1] = '\0';
    fflush(stdin);
    fprintf(Funcionario_arquivo, "\nNOME COMPLETO....."
"%s",nome_Funcionario[SIZE]);
    gotoxy(6,10);
    printf("DATA DE NASC.....");
    gotoxy(27,10);
    scanf("%s",&data_nasc_Funcionario[SIZE]);
    fflush(stdin);
    fprintf(Funcionario_arquivo, "\nDATA DE NASC....."
"%s",data_nasc_Funcionario[SIZE]);
    gotoxy(6,11);
    printf("CPF.....");
    gotoxy(27,11);
    scanf("%s",&cpf_Funcionario[SIZE]);
    fflush(stdin);

```

```

                                fprintf(Funcionario_arquivo,  "\nCPF.....:"
"%s",cpf_Funcionario[SIZE]);
    gotoxy(6,12);
    printf("TELEFONE.....:");
    gotoxy(27,12);
    scanf("%s",&telefone_Funcionario[SIZE]);
    fflush(stdin);
                                fprintf(Funcionario_arquivo,  "\nTELEFONE.....:"
"%s",telefone_Funcionario[SIZE]);
    gotoxy(6,13);
    printf("ENDERECO.....:");
    gotoxy(27,13);
    fgets(endereco_Funcionario[SIZE],200,stdin);
    endereco_Funcionario[SIZE][strlen(endereco_Funcionario[SIZE])-1] =
'\0';
    fflush(stdin);
                                fprintf(Funcionario_arquivo,  "\nENDERECO.....:"
"%s",endereco_Funcionario[SIZE]);
    gotoxy(6,14);
    printf("CIDADE.....:");
    gotoxy(27,14);
    fgets(cidade_Funcionario[SIZE],20,stdin);
    cidade_Funcionario[SIZE][strlen(cidade_Funcionario[SIZE])-1] = '\0';
    fflush(stdin);
                                fprintf(Funcionario_arquivo,  "\nCIDADE.....:"
"%s",cidade_Funcionario[SIZE]);
    gotoxy(6,15);
    printf("ESTADO.....:");
    gotoxy(27,15);
    fgets(estado_Funcionario[SIZE],50,stdin);
    estado_Funcionario[SIZE][strlen(estado_Funcionario[SIZE])-1] = '\0';
    fflush(stdin);
                                fprintf(Funcionario_arquivo,  "\nESTADO.....:"
"%s",estado_Funcionario[SIZE]);
    gotoxy(6,16);
    printf("CEP.....:");
    gotoxy(27,16);
    scanf("%s",&cep_Funcionario[SIZE]);
    fflush(stdin);

```



```

                                fprintf(Funcionario_arquivo,  "\nCEP.....:"
"%s",cep_Funcionario[SIZE]);
    gotoxy(6,17);
    printf("FUNCAO.....:");
    gotoxy(27,17);
    fgets(funcao_Funcionario[SIZE],50,stdin);
    funcao_Funcionario[SIZE][strlen(funcao_Funcionario[SIZE])-1] = '\0';
    fflush(stdin);

                                fprintf(Funcionario_arquivo,  "\nFUNCAO.....:"
"%s",funcao_Funcionario[SIZE]);
    gotoxy (18,19);
    mudacor(BRANCO,VERDE);
    printf("FUNCIONARIO CADASTRADO COM SUCESSO!!!");
    mudacor(BRANCO,PRETO);
    fclose(Funcionario_arquivo);
    sleep (1);
    menuGerente();
}
}
void consultaFuncionario()
{
    system("cls"); // limpeza de tela
    mudacor(BRANCO,PRETO);
    bordas();
    gotoxy (31,3);
    mudacor(BRANCO,AZUL);
    printf (" CONSULTA FUNCIONARIO ");
    gotoxy (31,5);
    mudacor(BRANCO, PRETO);
    char C_Funcionarios [1000];
    FILE *Funcionario_arquivo;
    Funcionario_arquivo = fopen("Funcionario.txt", "r");
    if (Funcionario_arquivo == NULL)
    {
        gotoxy(23,15);
        mudacor(BRANCO,VERMELHO);
        printf("NAO FOI POSSIVEL CRIAR O ARQUIVO");
        getchar();
    }
}

```

```

while (fgets(C_Funcionarios,1000,Funcionario_arquivo)!= NULL)
{
    printf("\t %s", &C_Funcionarios);
}
fclose(Funcionario_arquivo);
getchar();
menuGerente();
}
void excluirfuncionario()
{
    system("cls"); // limpeza de tela
    mudacor(BRANCO,PRETO);
    bordas(); // APLICANDO BORDAS
    gotoxy (33,3);
    mudacor(BRANCO, AZUL);
    printf (" EXCLUIR FUNCIONARIO ");
    mudacor(BRANCO,PRETO);
    mudacor(PRETO,AMARELO);
    gotoxy (28,10);
    printf("AREA EM DESENVOLVIMENTO!!! ");
    gotoxy (25,12);
    printf("EM BREVE ATUALIZACAO DO SOFTWARE");
    mudacor(BRANCO,PRETO);
    gotoxy(20,24);
    getchar();
    menuGerente();
}
void editarfuncionario()
{
    system("cls"); // limpeza de tela
    mudacor(BRANCO,PRETO);
    bordas(); // APLICANDO BORDAS
    gotoxy (33,3);
    mudacor(BRANCO, AZUL);
    printf (" EDITAR FUNCIONARIO ");
    mudacor(BRANCO,PRETO);
    mudacor(PRETO,AMARELO);
    gotoxy (28,10);

```

```

printf("AREA EM DESENVOLVIMENTO!!! ");
gotoxy (25,12);
printf("EM BREVE ATUALIZACAO DO SOFTWARE");
mudacor(BRANCO,PRETO);
gotoxy(20,24);
getchar();
menuGerente();
}
void sair()
{
    system ("cls"); // LIMPAR TELA
    mudacor(BRANCO,PRETO);
    bordas(); // APLICANDO BORDAS
    mudacor(BRANCO,AZUL);
    gotoxy (25,6); // POSICIONAR LINHA E COLUNA
    printf ("OBRIGADO POR UTILIZAR O SISTEMA!");
    gotoxy (13,10); // POSICIONAR LINHA E COLUNA
    mudacor (VERDE,PRETO);
    printf (" O SISTEMA FOI DESENVOLVIDO A FINS DE APRESENTACAO
DO PIM");
    gotoxy (32,13);
    mudacor(BRANCO,PRETO);
    printf ("AUTORES DO SISTEMA :");
    gotoxy (25,15);
    printf ("JONATHAN SOUZA MOYANO....RA G081960 ");
    gotoxy (25,16);
    printf ("BRUNA LINO MOYANO.....RA G095057");
    gotoxy (25,17);
    printf ("LUIZ FELIPE LUZ SANTOS...RA N029HA7");
    gotoxy (25,18);
    printf ("HELOISA DOS SANTOS COSTA.RA N029447");
    getchar();
    exit (0); // FECHAR O SISTEMA
}

```

## **8 CONCLUSÃO**

O objetivo da CTech é ser referência na prestação de serviços de suporte e manutenção de equipamentos com maior eficiência, segurança e agilidade, com qualidade e técnicas, seguindo padrões de processos para que o prazo de conclusão e entrega seja respeitado.

Nossa missão é atuar com profissionalismo e transparência oferecendo nossos serviços de maneira personalizada.

## REFERÊNCIAS

BERTELI, Hamilton. **Ética Digital**:: como equilibrar os riscos e recompensas da inovação digital. 2016. Disponível em: <https://canaltech.com.br/mercado/etica-digital-como-equilibrar-os-riscos-e-recompensas-da-inovacao-digital/>. Acesso em: 16 nov. 2022.

FABRIS BATTISTI, Júlio Cesar. **TCP/IP: Compartilhando a Conexão Internet**. 2020. Disponível em: [http://www.linhadecodigo.com.br/artigo/603/tutorial-de-tcp\\_ip-parte16-compartilhando-a-conexao-internet.aspx](http://www.linhadecodigo.com.br/artigo/603/tutorial-de-tcp_ip-parte16-compartilhando-a-conexao-internet.aspx). . Acesso em: 25 nov. 2022.

IBERDROLA, Iberdrola. **O QUE É A ÉTICA DIGITAL Ética na Internet**: : a chave do futuro digital. 2019. Disponível em: <https://www.iberdrola.com/compromisso-social/etica-na-internet>. Acesso em: 25 nov. 2022.

MILLS, Matt. **Escolhendo a fibra óptica certa para conectar dois switches**. 2021. Disponível em: <https://itigic.com/pt/choosing-right-fiber-optic-to-connect-twoswitches/>. Acesso em: 25 nov. 2022.

NOLETO, Cairo. **Topologias de rede**: o que são e quais os tipos? . 2020. Disponível em: <https://blog.betrybe.com/tecnologia/topologias-de-rede/>. Acesso em: 25 nov. 2022.

PERIM, Bruno . **A Revolução das Startups**: O Novo Mundo do Empreendedorismo de Alto Impacto . Disponível em: . Acesso em: 25 nov. 2022.