

**UNIP – UNIVERSIDADE PAULISTA**

**PROJETO INTEGRADO MULTIDISCIPLINAR – PIM II**  
**CURSO SUPERIOR DE TECNOLOGIA DE ANALISE E DESENVOLVIMENTO DE**  
**SISTEMAS**

**LUIZ CARLOS DE ARAÚJO MACHADO  
GABRIEL NATAN MAGALHÃES BENTO DE MEDEIROS  
KAIO HENRIQUE ALVES DA SILVA**

**SGCOE:**  
Sistema de Gestão Choveu? Onde Encheu?

**SANTOS SP  
2022**

**LUIZ CARLOS DE ARAÚJO MACHADO**  
**GABRIEL NATAN MAGALHÃES BENTO DE MEDEIROS**  
**KAIO HENRIQUE ALVES DA SILVA**

**SGCOE:**

Sistema de Gestão Choveu? Onde Encheu?

Projeto de integração multidisciplinar  
PIM apresentada como exigência para  
a conclusão de semestre, junto à  
Universidade Paulista UNIP.

Orientador: Prof. Dr. João Carlos  
Martins

**SANTOS SP**  
**2022**

CIP - Catalogação na Publicação

Machado, Luiz Carlos de Araújo

SGCOE: Sistema de Gestão Choveu? Onde Encheu? / Luiz Carlos de Araújo Machado, Gabriel Natan Magalhães Bento de Medeiros, Kaio Henrique Alves da Silva. - 2022.

68 f. : il. color

Trabalho de Conclusão de Curso (Graduação) apresentado ao Instituto de Ciência Exatas e Tecnologia da Universidade Paulista, Santos, 2022.

Área de Concentração: ..

Orientador: Prof. Dr. João Carlos Martins.

1. Desenvolvimento. 2. Startup. 3. Rede. 4. Gerenciador. I. Medeiros, Gabriel Natan Magalhães Bento de. II. Silva, Kaio Henrique Alves da. III. Martins, João Carlos (orientador). IV. Título.

Elaborada pelo Sistema de Geração Automática de Ficha Catalográfica da Universidade Paulista  
com os dados fornecidos pelo(a) autor(a).

**LUIZ CARLOS DE ARAÚJO MACHADO**  
**GABRIEL NATAN MAGALHÃES BENTO DE MEDEIROS**  
**KAIO HENRIQUE ALVES DA SILVA**

**SGCOE:**

Sistema de Gestão Choveu? Onde Encheu?

Projeto de integração multidisciplinar  
PIM apresentada como exigência para  
a conclusão de semestre, junto à  
Universidade Paulista UNIP.

Orientador: Prof. Dr. João Carlos  
Martins

Aprovado em:

**BANCA EXAMINADORA**

\_\_\_\_ / \_\_\_\_

Prof. Nome do Professor  
Universidade Paulista – UNIP

\_\_\_\_ / \_\_\_\_

Prof. Nome do Professor  
Universidade Paulista – UNIP

\_\_\_\_ / \_\_\_\_

Prof. Nome do Professor  
Universidade Paulista – UNIP

## **RESUMO**

No cotidiano de uma empresa, surgem diversas tarefas ou projetos que precisam ser planejadas, desenvolvidas e avaliadas de maneira eficiente e produtiva. A gestão responsável precisa pensar numa série de situações para que tal empreendimento obtenha sucesso, como, por exemplo, desenvolvimento do conteúdo, avaliação de desempenho, delegação de tarefas, entre outros, ou seja, é necessário organizar todas essas ações para se ter um controle claro dos processos existentes dentro de uma empresa, fazendo ser necessário o uso de um gerenciador de tarefas que é uma excelente ferramenta para centralizar vários processos. De acordo a pequena empresa Choveu? Onde Encheu? que consiste em ser uma startup sobre alagamentos na cidade de São Vicente, contratou a Devcoast que é grupo de analistas de sistemas para projetar a infra estrutura de rede e desenvolver um gerenciador de tarefas em linguagem C, já que a empresa está passando por um processo de aumento de suas atividades. Neste projeto integrado II, apresentaremos a proposta de infra estrutura de rede e mostrar em detalhes todas as etapas do processo de desenvolvimento do gerenciador de tarefas SGCOE.

Palavras-chave: Desenvolvimento. Startup. Rede. Gerenciador.

## **ABSTRACT**

In the daily life of a company, several tasks or projects arise that need to be planned, developed and evaluated in an efficient and productive way. Responsible management needs to think about a series of situations for such an undertaking to be successful, such as, for example, content development, performance evaluation, task delegation, among others, that is, it is necessary to organize all these actions to have control of existing processes within a company, making it necessary to use a task manager, which is an excellent tool to centralize several processes. According to the small company Did it rain? Where did it fill? which consists of being a startup about flooding in the city of São Vicente, hired Devcoast, which is a group of systems analysts, to design the network infrastructure and develop a task manager in C language, since the company is going through a process to increase their activities. In this integrated project II, we will present the proposed network infrastructure and show in detail all stages of the SGCOE task manager development process.

**Keywords:** Manager. Development. startup. Network.

## LISTA DE ILUSTRAÇÕES

Figura 1: Logo COE.....	11
Figura 2: Logo COE e logo Devcoast.....	11
Figura 3: Logo Devcoast.....	18
Figura 4: Infraestrutura de rede.....	19
Figura 5: Logo SGCOE.....	20
Figura 6: Fluxograma tela de login.....	21
Figura 7: Fluxograma tela perfil padrão.....	22
Figura 8: Fluxograma tela perfil gerente.....	23
Figura 9: Fluxograma tela perfil colaborador.....	24
Figura 10: Bibliotecas.....	25
Figura 11: Estrutura de dados.....	26
Figura 12: Bloco da função excluir dados.....	27
Figura 13: Função geradorId.....	27
Figura 14: Função lerString.....	28
Figura 15: Menu switch case.....	29
Figura 16: Bloco da função pesquisar.....	29
Figura 17: Função main.....	30
Figura 18: Tela inicial.....	30
Figura 19: Tela de login.....	31
Figura 20: Tela de login acesso negado.....	31
Figura 21: Tela de login após 3 tentativas.....	32
Figura 22: Tela de login logado com sucesso.....	32
Figura 23: Menu perfil master.....	33
Figura 24: Menu perfil master escolha inválida.....	34
Figura 25: Formulário de cadastro de gerente.....	34
Figura 26: Lista de cadastro de gerente.....	35
Figura 27: Menu de pesquisa.....	36
Figura 28: Menu de pesquisar por nome.....	37
Figura 29: Menu de pesquisa cadastro encontrado.....	37
Figura 30: Menu de pesquisa excluir cadastro.....	38
Figura 31: Menu de pesquisa excluído com sucesso.....	38
Figura 32: Menu perfil gerente.....	39

Figura 33: Formulário de cadastro de colaborador.....	40
Figura 34: Tela atribuir tarefas.....	40
Figura 35: Atribuindo tarefa.....	41
Figura 36: Nível de prioridade.....	41
Figura 37: Nível de prioridade escolha inválida .....	42
Figura 38: Data de entrega.....	42
Figura 39: Data de entrega prazo indefinido .....	42
Figura 40: Tarefa atribuída com sucesso.....	43
Figura 41: Tela monitorar tarefas .....	44
Figura 42: Menu monitorar tarefas .....	44
Figura 43: Monitorarando tarefas .....	45
Figura 44: Adicionando comentário.....	46
Figura 45: Excluindo tarefa .....	47
Figura 46: Tela monitorar tarefas atualizado.....	48
Figura 47: Menu cliente .....	48
Figura 48: Formlulário de cadastro de cliente.....	49
Figura 49: Lista de cadastro de cliente .....	49
Figura 50: Exportando relatório.....	50
Figura 51: Arquivo txt .....	51
Figura 52: Relatorio Arquivo txt.....	52
Figura 53: Bloco de códigos 1 da função criptografia .....	55
Figura 54: Bloco de códigos 2 da função criptografia .....	56
Figura 55: Bloco de códigos 3 da função criptografia .....	57
Figura 56: Bloco de códigos 3 da função criptografia .....	57
Figura 57: Bloco de códigos 3 da função criptografia .....	58
Figura 58: Bloco de códigos 4 da função criptografia .....	58
Figura 59: Bloco de códigos 5 da função criptografia .....	59
Figura 60: Tela de descriptografar.....	59
Figura 61: Arquivo relatorioDescript.txt.....	60
Figura 62: Arquivo descriptografado.....	60
Figura 63: Tela login colaborador.....	61
Figura 64: Tela menu colaborador.....	61
Figura 65: Tela tarefas colaborador.....	62
Figura 66: Tela concluindo tarefa.....	63

Figura 67: Tela tarefa concluída..... 63

## SUMÁRIO

1	<b>INTRODUÇÃO .....</b>	11
2	<b>OBJETIVO GERAL.....</b>	13
2.1	<b>Objetivos específicos .....</b>	13
3	<b>REFERENCIAL TEÓRICO .....</b>	15
3.1	<b>Gerenciador de tarefas .....</b>	15
4	<b>DESENVOLVIMENTO .....</b>	17
4.1	<b>Linguagem C .....</b>	17
4.1.1	Por que utilizar C?.....	17
4.2	<b>Devcoast .....</b>	17
4.3	<b>Infraestrutura de Rede.....</b>	19
4.4	<b>Sistema de Gestão Choveu? Onde Encheu? .....</b>	20
4.4.1	Fluxograma macro do processo.....	21
4.4.2	Códigos .....	25
4.4.3	Demostraçao do programa.....	30
4.4.4	Cadastrar gerente .....	34
4.4.5	Lista de cadastro .....	35
4.4.6	Pesquisar .....	35
4.4.7	Login e Sair .....	38
4.4.8	Perfil Gerente .....	39
4.4.8.1	Cadastrar colaborador.....	39
4.4.8.2	Atribuir tarefas .....	40
4.4.8.3	Monitorar tarefas .....	44
4.4.8.4	Cliente .....	48
4.4.8.5	Exportar relatório.....	50
4.4.8.6	Criptografia .....	53
4.4.8.6.1	O que é criptografia de dados? .....	53
4.4.8.6.2	Quais são os tipos de criptografia? .....	53
4.4.8.6.2.1	Criptografia simétrica .....	53
4.4.8.6.2.2	Criptografia Assimétrica .....	54

4.4.8.6.2.3	Funções de Hash .....	54
4.4.8.6.2.4	Assinaturas digitais .....	55
4.4.8.7	Criptografia SGCOE.....	55
4.4.9	Perfil Colaborador .....	61
4.4.9.1	Tarefas pendentes .....	62
<b>5</b>	<b>CONSIDERAÇÕES FINAIS.....</b>	<b>65</b>
	<b>REFERÊNCIAS.....</b>	<b>66</b>

## 1 INTRODUÇÃO

COE (Choveu? Onde Encheu?) é uma startup criada na cidade de São Vicente com a finalidade de ajudar com os problemas causados por alagamentos e enchentes por meio de um aplicativo mobile (Machado, 2022).

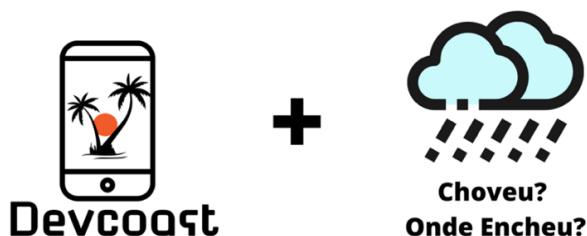
Figura 1: Logo COE.



Fonte: Próprios autores, 2022.

Com o sucesso do aplicativo a startup recebeu um aporte financeiro que possibilitou a aquisição de computadores modestos, mesas, cadeiras, aluguel, contratação de alguns funcionários, e outros equipamentos, porém nada muito sofisticado, então por consequência a empresa contratou nossa equipe de analistas de sistemas que chamamos de Devcoast para projetar a infra estrutura de rede e desenvolver um software para gerenciar as atividades da startup.

Figura 2: Logo COE e logo Devcoast.



Fonte: Próprios autores, 2022.



## 2 OBJETIVO GERAL

Desenvolver um programa que permita realizar o controle de todos os processos envolvidos para o gerenciamento de uma startup com base nas disciplinas Linguagem e Técnicas De Programação, Engenharia de Software I, Fundamentos Redes Dados e Comunicação, Matemática para Computação.

### 2.1 Objetivos específicos

- Desenvolver e aplicar os conhecimentos adquiridos em sala de aula;
- Desenvolver algoritmos e os respectivos fluxogramas;
- Explorar e utilizar ferramentas computacionais para desenvolvimento e elaboração de diagramas de fluxo de software;
- Uso da matemática em programas de computador;
- Debugar o código passo a passo e investigar o funcionamento do mesmo;
- Fomentar o hábito de trabalho em equipe e execução de projetos envolvendo múltiplas disciplinas.



### 3 REFERENCIAL TEÓRICO

#### 3.1 Gerenciador de tarefas

De acordo com (Pires, 2018) no cotidiano de uma empresa, surgem diversas tarefas ou projetos que precisam ser planejadas, desenvolvidas e avaliadas de maneira eficiente e produtiva. A gestão responsável precisa pensar numa série de situações para que tal empreendimento obtenha sucesso, como, por exemplo, desenvolvimento do conteúdo, avaliação de desempenho, delegação de tarefas, publicação do conteúdo, definição de indicadores de desempenho, entre outros. Com a necessidade de organizar todas essas ações, o gerenciador de tarefas é uma excelente ferramenta para centralizar vários processos, se tornando assim uma ferramenta essencial para a gestão que busca otimizar os processos da empresa.

Para (Proj4me, 2021) um gerenciador de tarefas é uma ferramenta (software) que permite que você organize as suas atividades dentro de uma empresa, priorizando as mais urgentes e monitorando o status de cada uma, até mesmo, programar alertas para que todos saibam exatamente o que deve ser feito diariamente, melhorando a gestão do tempo e qualidade das entregas. Segundo o autor, ter uma ferramenta que facilita essa organização é essencial para garantir um fluxo adequado de atividades e ter todas as tarefas realizadas dentro do prazo estabelecido.

(Espinha, 2022) comenta que entender para que serve a gestão de tarefas num empreendimento é crucial para que as vantagens de praticá-la sejam tão perceptíveis quanto os seus desafios. O autor listou alguns pontos importantes sobre as vantagens de utilizar o gerenciador de tarefas:

- Otimização de produtividade e aumento de performance permitindo que o colaborador ajuste as atividades em função do tempo que ele tem disponível;
- Vantagem direta da gestão de tarefas é a priorização de tarefas muito importantes e a delegação de tarefas menos importantes;
- Conhecimento claro e cumprimento de prazos conhecidos permitindo que sejam organizados de forma realista e possível;
- Melhora na comunicação e no clima organizacional.



## 4 DESENVOLVIMENTO

### 4.1 Linguagem C

Criada pelo cientista da computação Dennis Ritchie, em 1972, a linguagem C foi derivada de outras duas: a BCPL e a Algol 68. Embora tenha sido pensada com o propósito exclusivo de ser uma linguagem de programação usada no desenvolvimento de uma nova versão do sistema operacional Unix, hoje é aplicada nos mais variados tipos de projeto. A linguagem C ainda é uma das mais populares do mercado de programação devido às diversas vantagens que apresenta (Noleto, 2022).

Para (Noleto, 2022) a linguagem C é considerada uma linguagem de alto nível e pode ser usada em diversos tipos de projeto, como a criação de aplicativos, sistemas operacionais, drivers, entre outros. O autor frisa que a linguagem C se tornou muito popular nos anos 80 e que é difícil encontrar arquiteturas para as quais não existam compiladores para a C, garantindo o seu elevado nível de portabilidade. Uma das grandes vantagens desta linguagem, segundo (Noleto, 2022), é a capacidade de gerar códigos rápidos, ou seja, com um tempo de execução baixo. Além disso, a programação em C é bastante simplificada, pois sua estrutura é simples e flexível.

Por fim, (Noleto, 2022) comenta que a popularidade da linguagem C foi muito significante ao ponto de influenciar diretamente a estrutura e sintaxe de outras, como C++, Objective C e C#.

#### 4.1.1 Por que utilizar C?

Para Devcoast desenvolver o programa em modo console utilizando a linguagem C é adequado com as necessidades exigidas, pois os computadores do cliente são bastante limitados, entretanto existem outras vantagens apontadas por nós em utilizar a linguagem C, como portabilidade, geração de código eficiente, simplicidade, confiabilidade, facilidade de uso e consistência de mercado.

### 4.2 Devcoast

Devcoast (Dev = developer e Coast = litoral) fundada em Santos litoral de São Paulo, por de 3 estudantes entusiastas da área de T.I (Tecnologia da Informação)

possui como sua principal missão prestar serviços de qualidade e de forma ágil utilizando as mais modernas ferramentas disponíveis no mercado. Nossa equipe é composta por profissionais qualificados que atendem todas as exigências de nossos clientes.

A seguir confira os principais serviços oferecidos:

- Serviços de TI: Gerenciamento completo do seu ambiente de TI, como planejamento de infraestrutura de rede;
- Soluções Customizadas: O cliente escolhe o projeto que melhor se encaixa a suas necessidades e implementamos;
- Licenciamento: Conte com os nossos especialistas para te ajudar a escolher o melhor licenciamento e compra de hardware;
- Desenvolvimento de aplicações Web, Windows e Android.

Figura 3: Logo Devcoast.



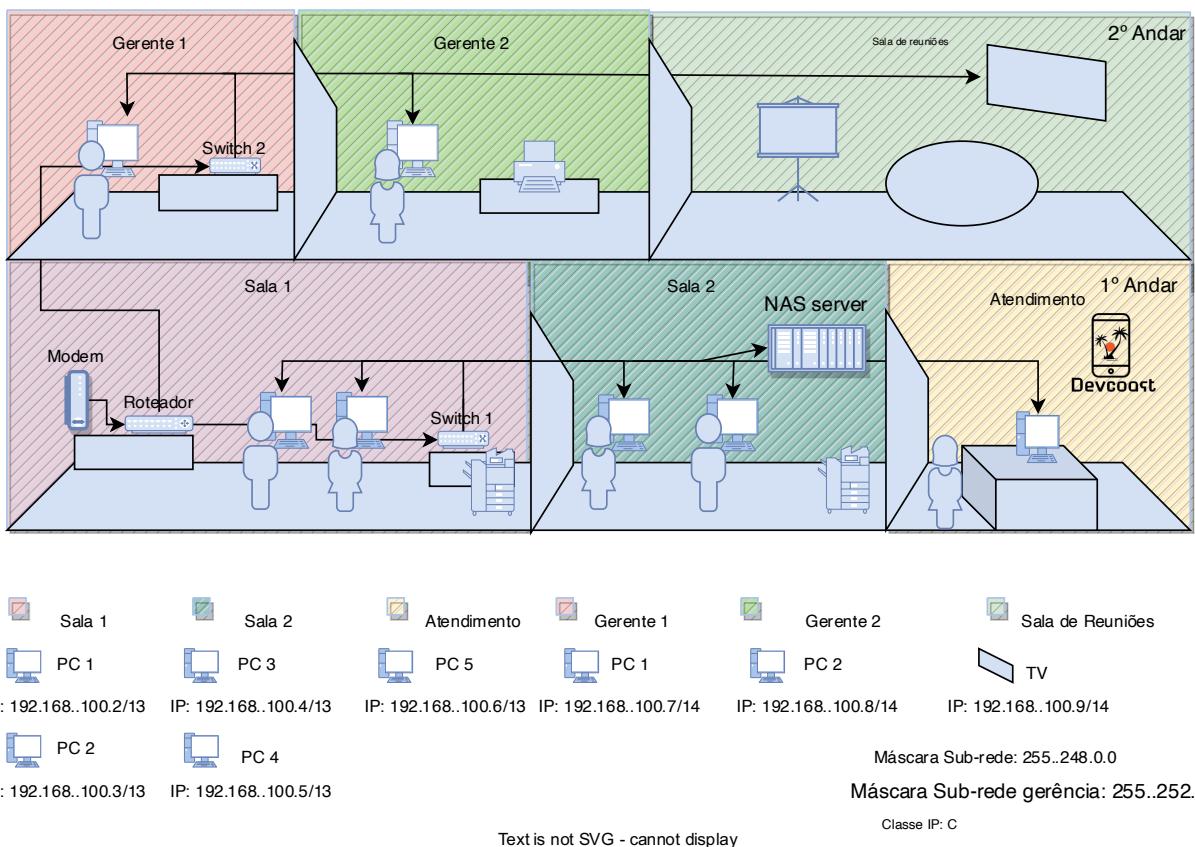
Fonte: Próprios autores, 2022.

### 4.3 Infraestrutura de Rede

A startup choveu? Onde Encheu? recentemente estava situada em uma única sala dentro da faculdade onde foi criada. Seus integrantes comunicavam-se pessoalmente uns com os outros, pois estavam em mesas lado a lado. Porém, devido a um aporte financeiro, a startup saiu desta única sala para dois conjuntos compostos de três salas cada, e em andares diferentes de um edifício comercial. Das seis salas, uma será destinada a reuniões, onde haverá uma TV conectada à rede para exibir apresentações. Os dois fundadores ficaram cada um com uma sala, no mesmo andar da sala de reuniões. No outro conjunto haverá uma sala para atendimento a clientes onde será locada uma recepcionista e nas outras duas salas haverá quatro funcionários, dois por sala.

Dado essa situação, a Devcoast desenvolveu o projeto de infra estrutura de rede para atender as necessidades do cliente.

Figura 4: Infraestrutura de rede.



Fonte: Próprios autores, 2022.

#### 4.4 Sistema de Gestão Choveu? Onde Encheu?

SGCOE (Sistema de Gestão Choveu? Onde Encheu?). No SGCOE o usuário encontra as seguintes ferramentas:

- Sistema de cadastro gerente, onde o usuário poderá criar perfis de gerência, no qual tem acesso a todas as funcionalidades do sistema;
- Sistema de login e senha, onde os dados do usuário são privados, limitando o acesso de pessoas não autorizadas;
- Sistema de atribuição de tarefas, onde o gerente pode atribuir e monitorar tarefas de cada colaborador de sua equipe;
- Sistema de cadastro de clientes;
- Exportar relatório em arquivo para o computador, contendo dados para tomada de decisão.

Figura 5: Logo SGCOE.



Fonte: Próprios autores, 2022.

#### 4.4.1 Fluxograma macro do processo

Nos fluxogramas a seguir aborda o processo e as particularidades individuais de cada perfil do programa de forma simples.

Figura 6: Fluxograma tela de login.

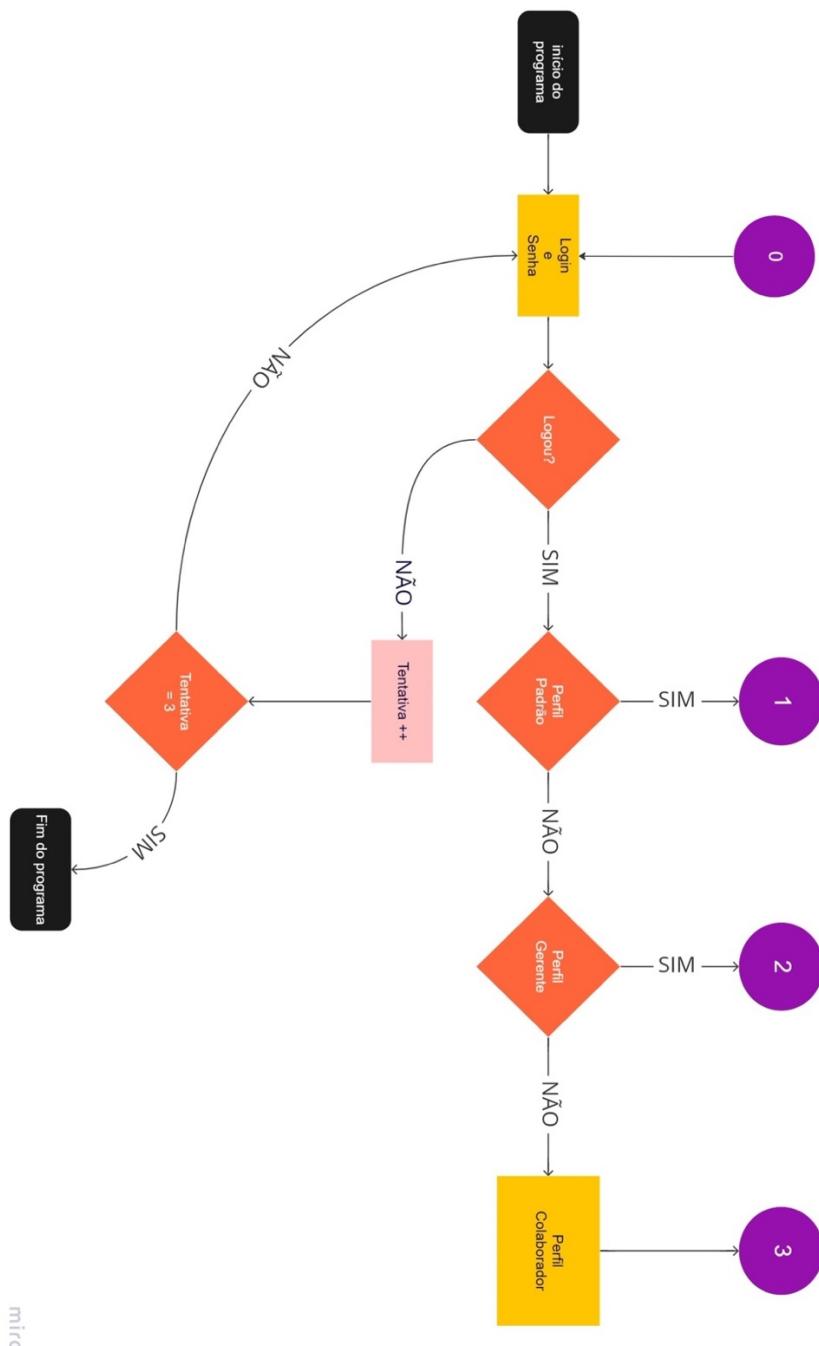
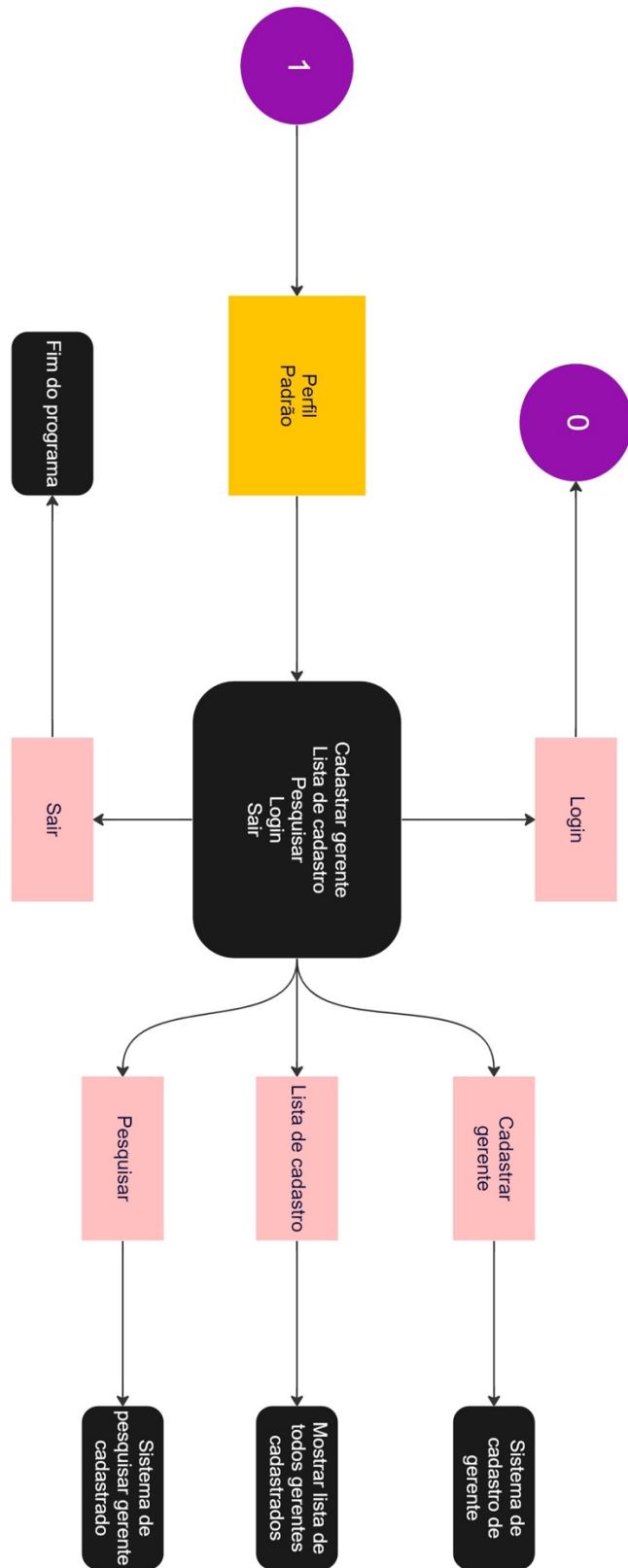


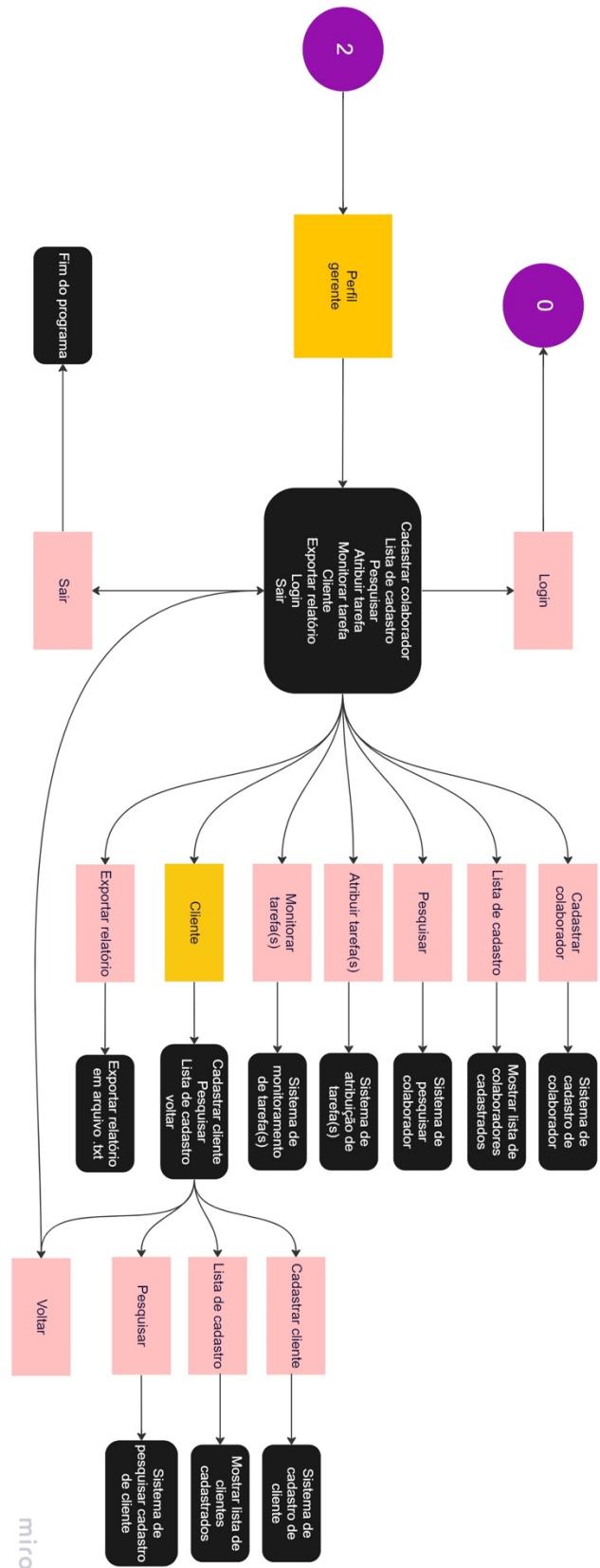
Figura 7: Fluxograma tela perfil padrão.



miro

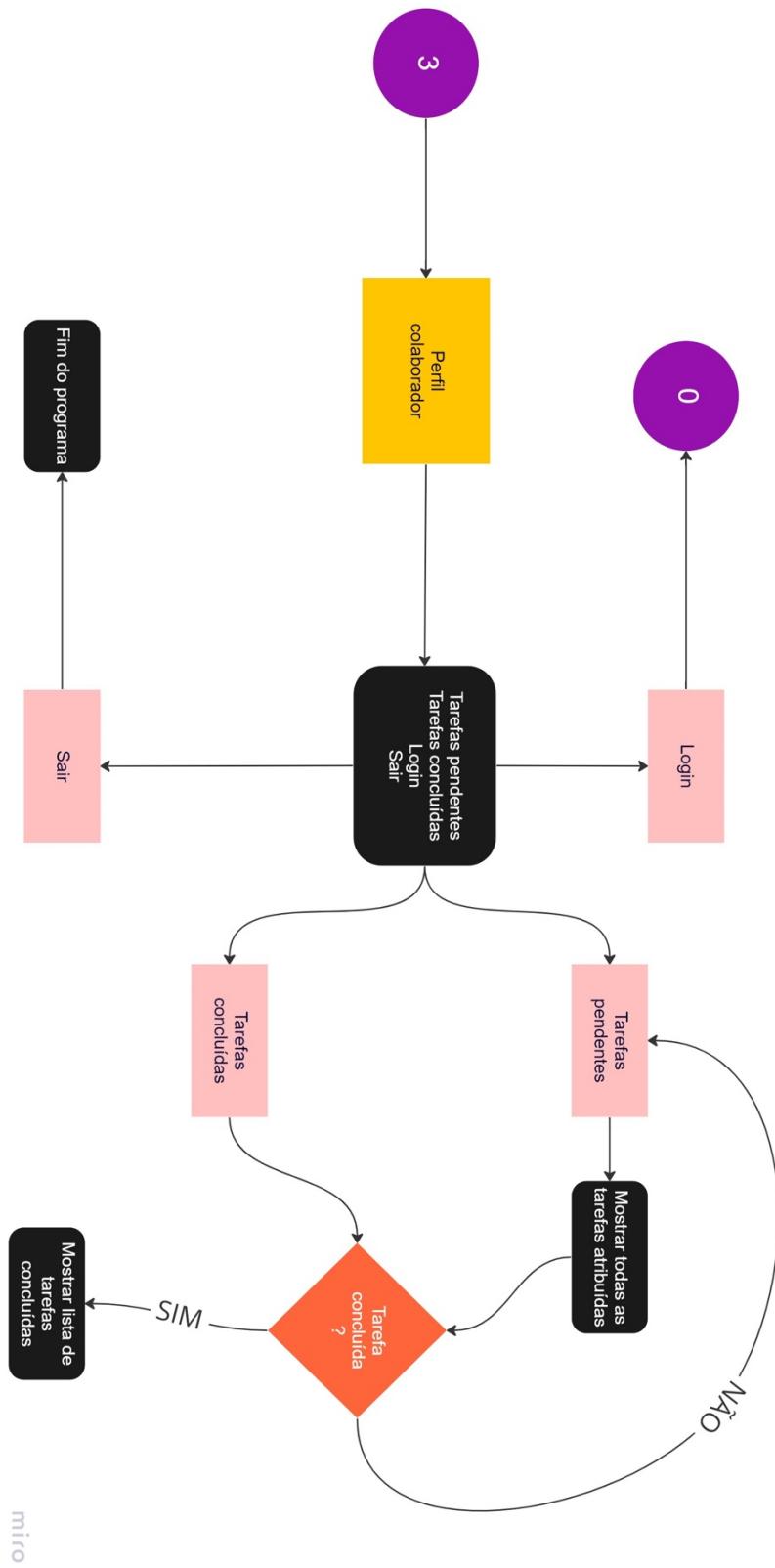
Fonte: Próprios autores, 2022.

Figura 8: Fluxograma tela perfil gerente.



Fonte: Próprios autores, 2022.

Figura 9: Fluxograma tela perfil colaborador.



Fonte: Próprios autores, 2022.

#### 4.4.2 Códigos

Para o desenvolvimento do programa SGCOE utilizamos a linguagem de programação C e o ambiente de desenvolvimento **Code::Blocks**, iremos apresentar e comentar a seguir alguns blocos de códigos que foram essenciais para o projeto.

Figura 10: Bibliotecas.

```

1  /*SISTEMA DE GERENCIAMENTO CHOVEU? ONDE ENCHEU?
2  SGCOE versao 0.1
3  22/09/2022 */
4
5  #include <stdio.h>
6  #include <stdlib.h>
7  #include <conio.h>
8  #include <conio.c>
9  #include <string.h>
10 #include <windows.h>
11 #include <time.h>
12 #include <stdbool.h>
13 #include <locale.h>
14
15 #define SIZE 300
16

```

Fonte: Próprios autores, 2022.

Durante a codificação foi necessário utilizar algumas bibliotecas para a elaboração do programa tais como, **windows.h** que nos permite utilizar algoritmos relacionados ao sistema operacional Windows, **time.h** que nos possibilita utilizar funções sobre o horário e **conio.c** que nos permite usar os comandos de cores para deixar a saída dos dados com um layout mais agradável para o usuário.

No desenvolvimento do código usamos algumas variáveis específicas que fazem parte da coluna central do sistema, tais como, a variável **linha**, **linha2**, **linha3** e **identificarColab** que são indicadores de índice para vetores. Também vale ressaltar a **struct dados\_1** que é o conjunto de variáveis de vários tipos, usados para armazenar dados de cada perfil, seja eles colaborador, gerente e padrão.

Figura 11: Estrutura de dados.

```

17 const char login[SIZE] = "Admin";
18 const char senha[SIZE] = "admin";
19 static int linha, linha2, linha3, identificarColab;
20 static char caractere;
21 int op;
22 int loginSenha();
23 struct dados_1
24 {
25     char nome[SIZE];
26     char email[SIZE];
27     char login[SIZE];
28     char senha[SIZE];
29     char ramal[SIZE];
30     char cargo[SIZE];
31     char titulo[SIZE][SIZE];
32     char descricao[SIZE][SIZE];
33     char prioridade[SIZE][SIZE];
34     char comentario[SIZE][SIZE];
35     int idComentario[SIZE];
36     int data[3][SIZE];
37     int dataConclusao[3][SIZE];
38     int hora[3][SIZE];
39     int prazo[SIZE];
40     int idTarefa[SIZE];
41     int id, quantTarefa, tarefasConcluida, tarefasPendente;
42     bool concluir[SIZE];
43 } gerente[SIZE], colaborador[SIZE];

```

Fonte: Próprios autores, 2022.

O trecho de código a seguir está localizado dentro de uma função chamada excluir dados, o código mostra algumas funções interessantes como a **memset** que utilizamos para limpar o conteúdo de um vetor e a função **strcpy** que copia os dados do tipo **string**. O algoritmo é simples, primeiro ele verifica qual cadastro o usuário solicitou a exclusão, depois ele limpa o conteúdo do vetor, já que todos os dados do sistema são armazenados em vetores e por fim copia os dados do índice da matriz posterior e atribui para o índice atual que foi limpo, isso acontece dentro de um laço de repetição, que termina quando ele realoca todos os dados da matriz.



A função **lerString** embora seja um algoritmo simples é de longe a função mais importante de todo o sistema, essa função é responsável pela entrada de dados do tipo **string**, ela consegue receber e armazenar palavra, frases e até pequenos textos em vetores, pois funções como **scanf** são limitadas com relação a receber dados do tipo **string**.

Figura 14: Função lerString.

```

317  char lerString(char palavra [SIZE])
318  {
319      static char letra;
320      static int i;
321
322      do {
323          letra=getch();
324          if(isprint(letra))
325          {
326              palavra[i]=letra;
327              printf("%s", &letra);
328              i++;
329          }
330          else if(letra==8&&i)
331          {
332              palavra[i]='\0';
333              i--;
334              printf("\b \b");
335          }
336      }while(letra!=13);
337      palavra[i]='\0';
338      i=0;
339      return palavra;
340 } //fim lerString

```

Fonte: Próprios autores, 2022.

O bloco de código a seguir mostra a estrutura **switch** utilizado para desenvolver menus de escolha ao longo do programa.



Ao final foram mais de 2000 linhas de códigos para desenvolver o algoritmo do SGCOE.

Figura 17: Função main.

```

2171 int main(void)
2172 {
2173     iniciarSistema();
2174 } //fim main
2175

```

Fonte: Próprios autores, 2022.

#### 4.4.3 Demostraçāo do programa

Após executar o programa na primeira tela é exibido o logo do sistema e uma mensagem para iniciar o programa.

Figura 18: Tela inicial.

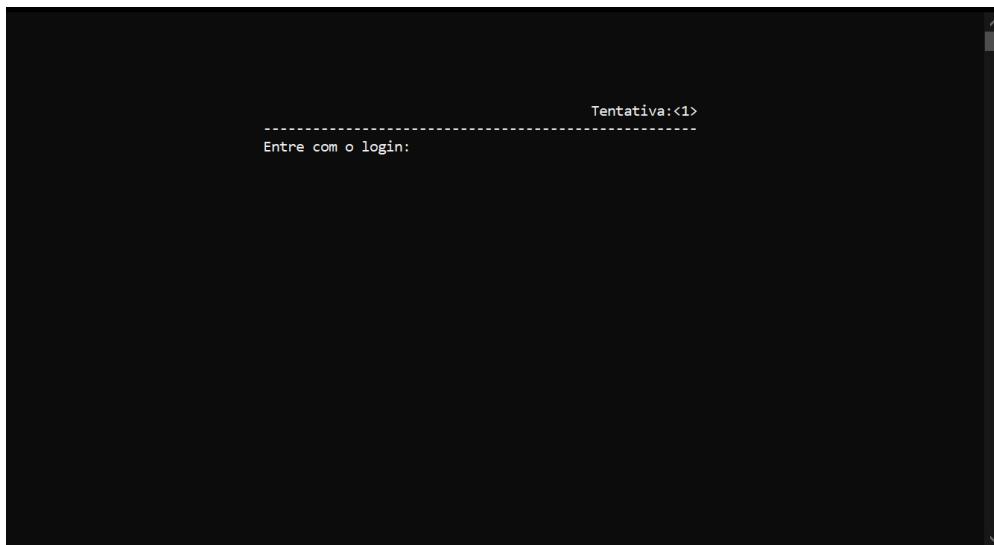


Fonte: Próprios autores, 2022.

Na mensagem sugere que o usuário tecle qualquer tecla para continuar ou digite “s” para encerrar o programa.

A próxima tela é a de acesso onde é pedido a senha e o login padrão do sistema definidos como: login “Admin” e Senha: “admin”, esse perfil é exclusivo e apenas a gestão da empresa tem acesso.

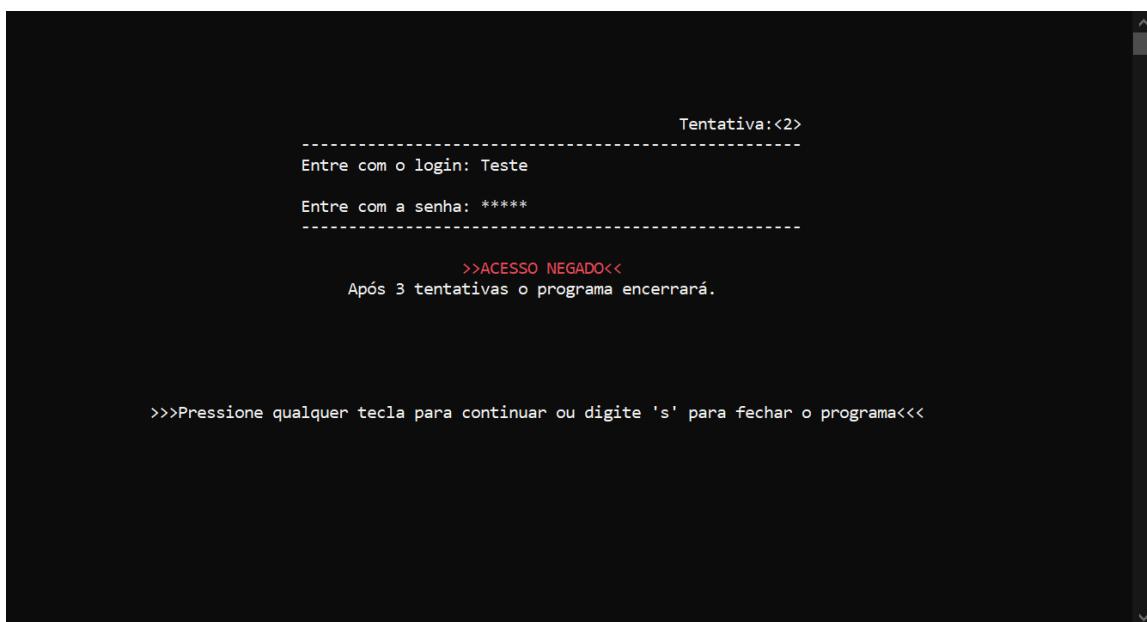
Figura 19: Tela de login.



Fonte: Próprios autores, 2022.

O usurário possui 3 tentativas de acesso, a quantidade de tentativas é exibida no canto superior da linha pontilhada e sempre que o usuário errar o acesso o sistema retornará uma mensagem de “Acesso negado”.

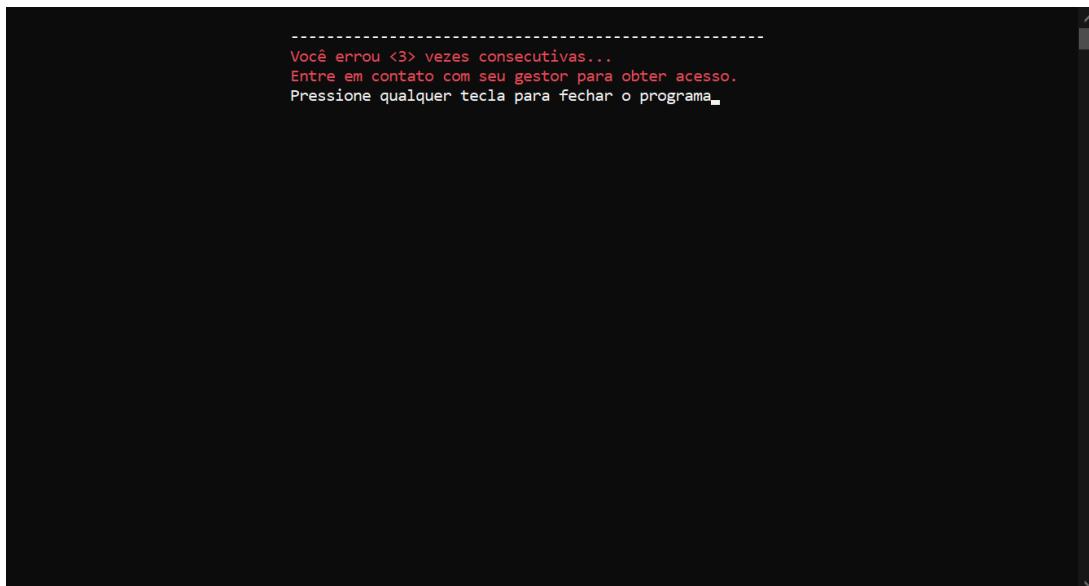
Figura 20: Tela de login acesso negado.



Fonte: Próprios autores, 2022.

Após esgotar a quantidade de tentativas, o programa exibe uma mensagem e encerrará.

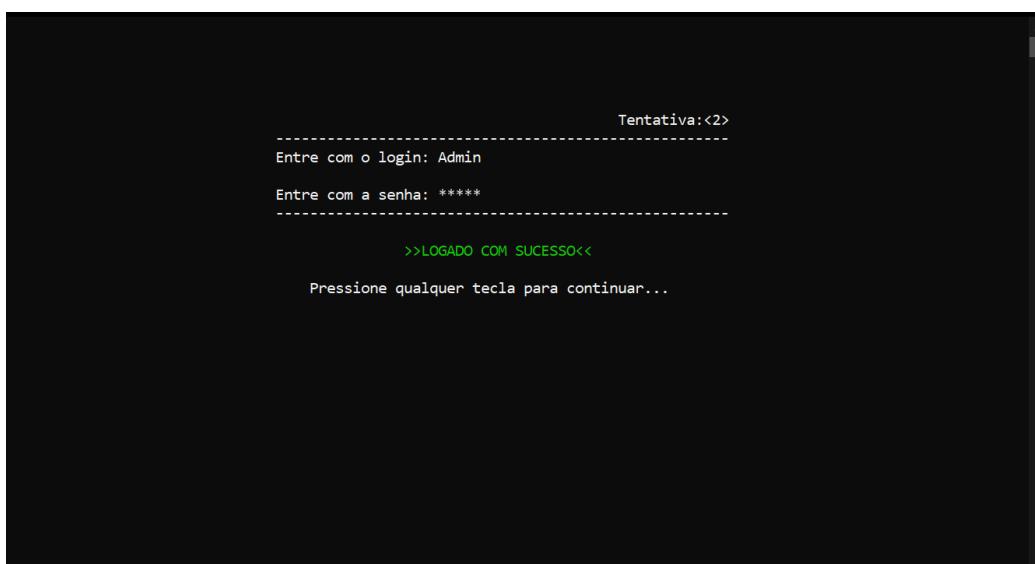
Figura 21: Tela de login após 3 tentativas.



Fonte: Próprios autores, 2022.

Ao validar o acesso o programa exibe a mensagem "Logado com sucesso". E libera o acesso ao perfil master.

Figura 22: Tela de login logado com sucesso.



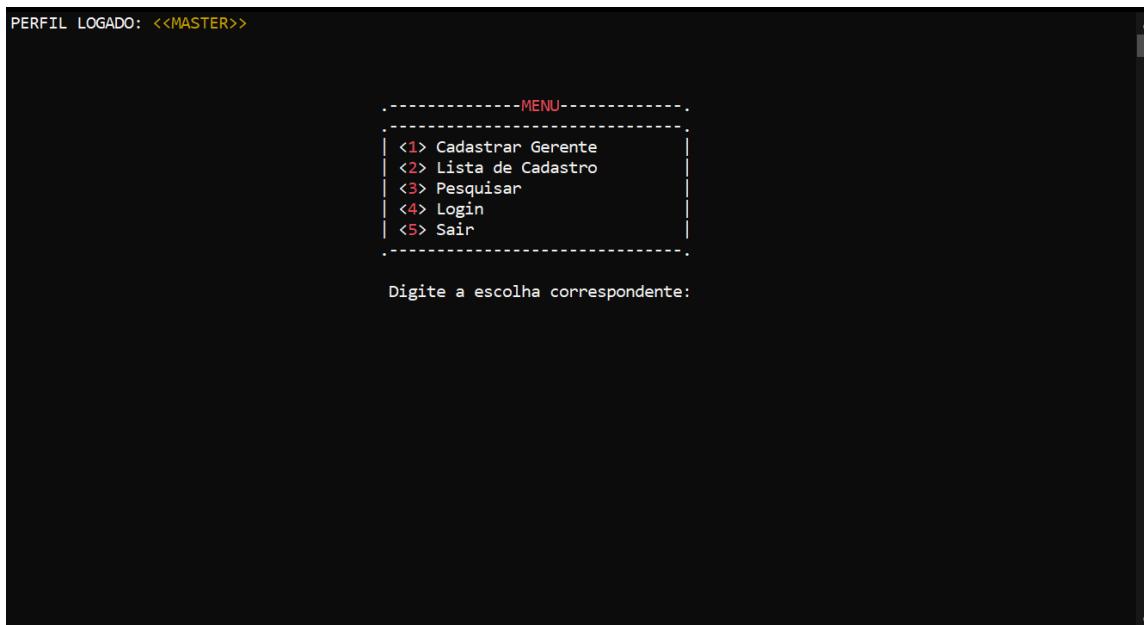
Fonte: Próprios autores, 2022.

Na tela do perfil master o programa exibe o menu contendo 5 opções validas:

- Cadastrar Gerente;
- Lista de cadastro;

- Pesquisar;
- Login;
- Sair.

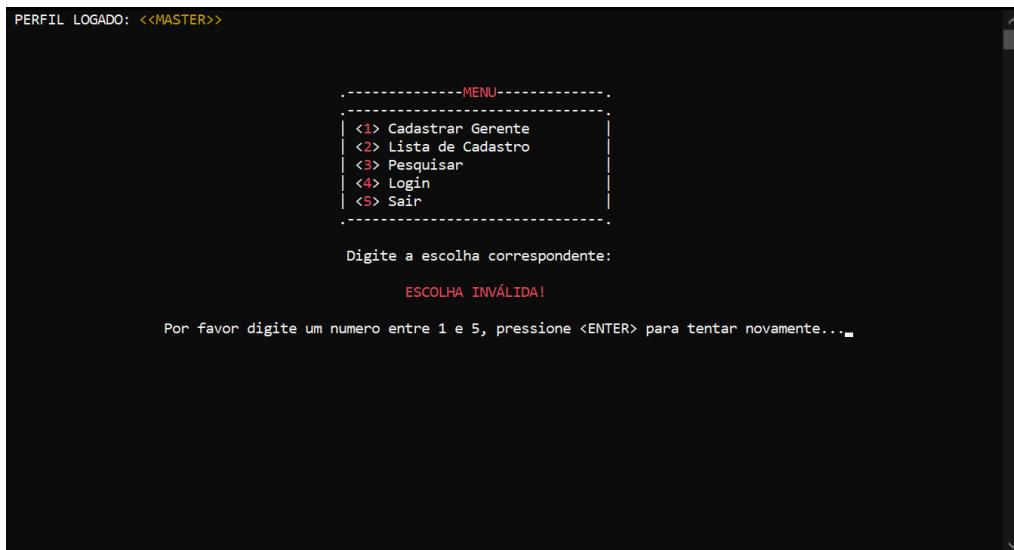
Figura 23: Menu perfil master.



Fonte: Próprios autores, 2022.

O usuário está limitado a escolha apenas entre as opções 1 e 5, mas caso o usuário fuja dessa regra o programa retornará uma mensagem de erro.

Figura 24: Menu perfil master escolha inválida.



Fonte: Próprios autores, 2022.

#### 4.4.4 Cadastrar gerente

Nesta opção o usuário pode cadastrar um perfil de gerência no qual terá total acesso do sistema. Para cadastrar o usuário deverá conceder todas as informações necessárias que o sistema pede, como nome, e-mail, ramal, login e senha.

Figura 25: Formulário de cadastro de gerente.

```

- FORMULÁRIO DE CADASTRO -
<<Após preencher um campo, pressione <ENTER> para seguir para o próximo>>

Nome completo: Gabriel Natan Medeiros
Email: Gabriel@email.com
Ramal: 1533
Cadastrar login: GabrielLogin
Cadastrar senha: GabrielSenha

CADASTRO REALIZADO COM SUCESSO!

Deseja continuar cadastrando?
<<Pressione <1> para continuar ou qualquer outra tecla para sair>>

```

Fonte: Próprios autores, 2022.

O formulário deverá ser preenchido corretamente, caso contrário o programa retornará uma mensagem invalidando o cadastro. Importante lembrar que o sistema

suporta até 300 perfis de gerentes cadastrados, numero mais do que suficiente para atendem a startup COE.

#### 4.4.5 Lista de cadastro

Nesta opção o usuário tem acesso a uma lista contendo todos os perfis cadastrados no sistema.

Figura 26: Lista de cadastro de gerente.

```
-LISTA DE CADASTRO(S)-

ID: 4879
Nome: Gabriel Natan Medeiros
Email: Gabriel@email.com
Ramal: 1533
Login: GabrielLogin
Senha: Gabrielsenha

Pressione qualquer tecla para continuar. . .
```

Fonte: Próprios autores, 2022.

Importante frisar que na primeira linha exibe um número de “ID”, esse é um número de identificação único para cada perfil independente qual seja (gerente ou colaborador), ou seja, cada cadastro é único, impossibilitando números de identificação idênticos. O ID é gerado por um algoritmo interno no sistema, não sendo possível o usuário modificar esse numero.

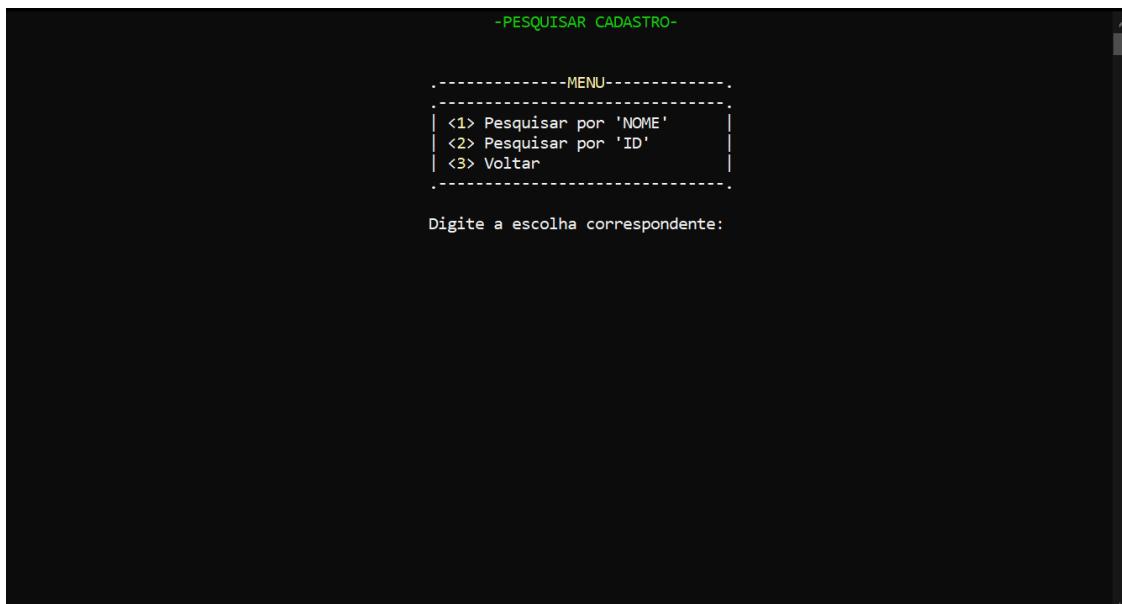
#### 4.4.6 Pesquisar

Nesta opção o sistema vai exibir uma tela exclusiva para pesquisa contendo o seguinte menu de opções:

- Pesquisar por nome;

- Pesquisar por ID;
- Voltar.

Figura 27: Menu de pesquisa.

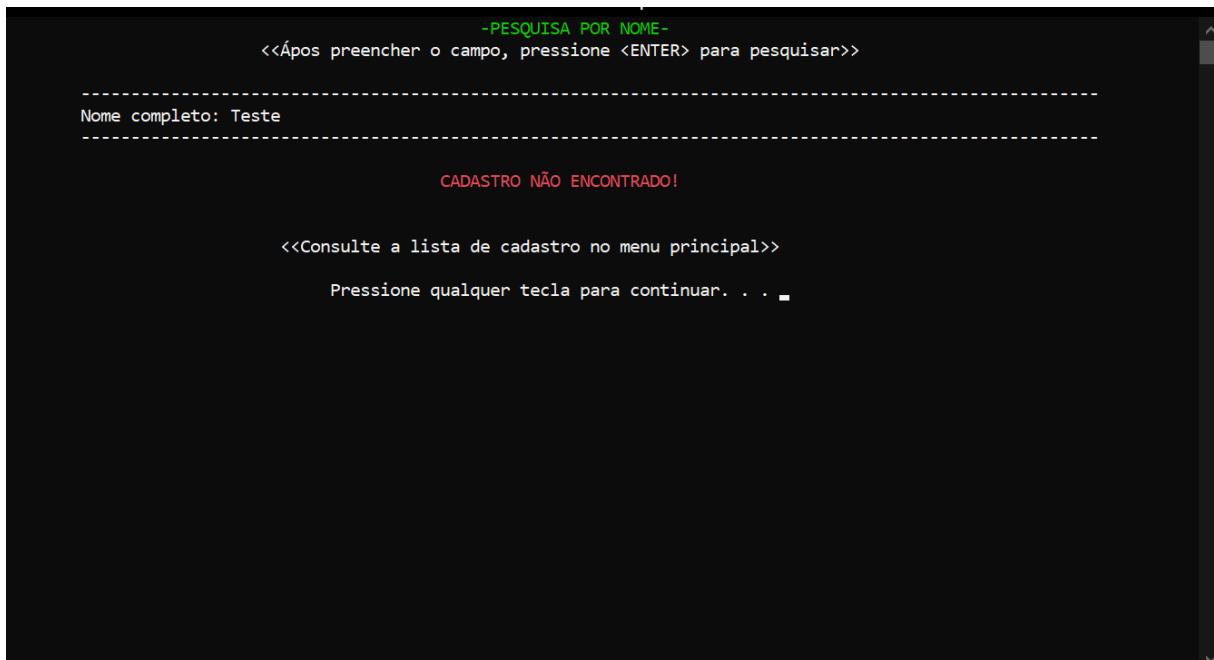


Fonte: Próprios autores, 2022.

Na opção “Pesquisar por nome” o usuário pode pesquisar perfis no sistema preenchendo o campo com o nome completo do cadastrado e na opção “Pesquisar por ID” utilizando o ID gerado pelo sistema, vale lembrar que esse ID é exibido na “Lista de cadastro” ver 4.4.5 Lista de cadastro. Entretanto, vale frisar que com o acesso “Master” a pesquisa buscará dados apenas de gerentes. E na opção “Voltar” o programa retornará para tela onde está o menu principal no perfil master.

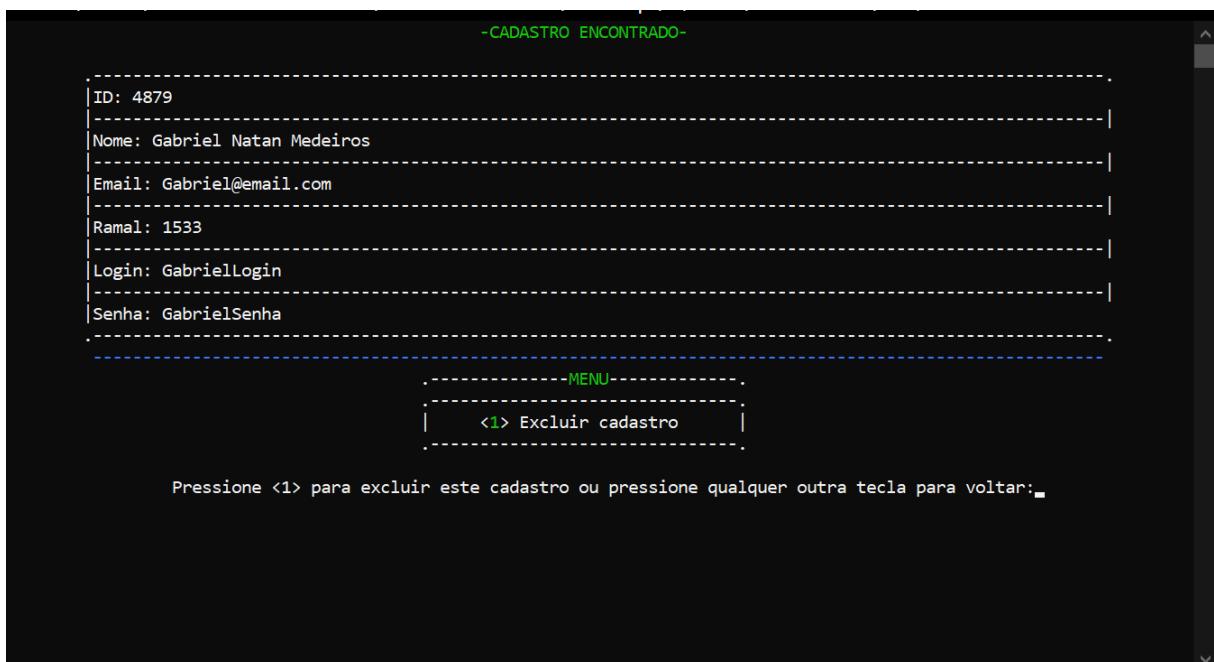
Caso o sistema de pesquisa não encontrar nenhum perfil, ele vai exibir uma mensagem de “Cadastro não encontrado”. Ao encontrar um perfil ele mostrará os dados e exibir um menu com a opção “Excluir cadastro”, essa opção excluirá o cadastro permanentemente do sistema, por esse motivo caso essa opção for escolhida será exibida uma mensagem de confirmação, veja as imagens a seguir.

Figura 28: Menu de pesquisar por nome.



Fonte: Próprios autores, 2022.

Figura 29: Menu de pesquisa cadastro encontrado.



Fonte: Próprios autores, 2022.

Figura 30: Menu de pesquisa excluir cadastro.

```
-CADASTRO ENCONTRADO-

ID: 4879
Nome: Gabriel Natan Medeiros
Email: Gabriel@email.com
Ramal: 1533
Login: GabrielLogin
Senha: GabrielSenha

----- MENU -----
<1> Excluir cadastro

Pressione <1> para excluir este cadastro ou pressione qualquer outra tecla para voltar:
<<ATENÇÃO ESTE CADASTRO SERÁ EXCLUÍDO PERMANENTEMENTE DO SISTEMA>>
<<Pressione <1> para confirmar ou pressione qualquer outra tecla para voltar>>
```

Fonte: Próprios autores, 2022.

Figura 31: Menu de pesquisa excluído com sucesso.

```
CADASTRO EXCLUÍDO COM SUCESSO!
Pressione qualquer tecla para continuar. . .
```

Fonte: Próprios autores, 2022.

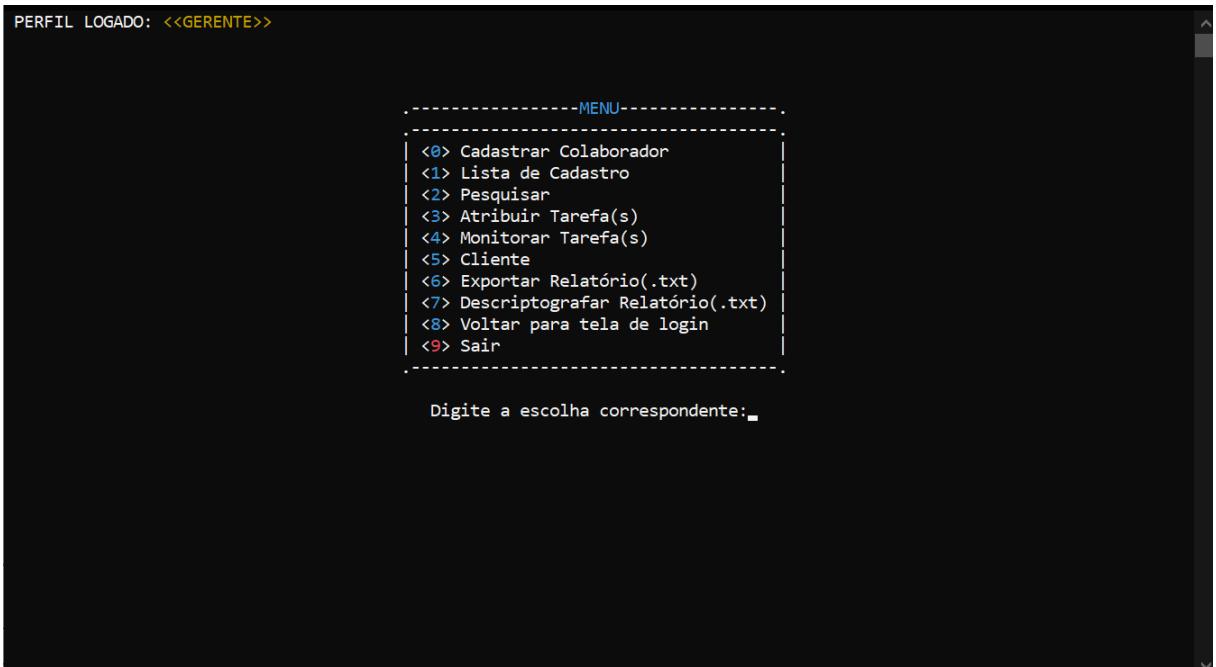
#### 4.4.7 Login e Sair

Na opção “Login” o programa retornará para tela de login inicial e na opção “Sair” o programa encerará.

#### 4.4.8 Perfil Gerente

No momento do cadastramento do perfil de gerente é exigido criar um login e senha no qual concede ao gerente acesso ao SGCOE. Após efetuar o login corretamente, o seguinte menu de opções é apresentado.

Figura 32: Menu perfil gerente.



Fonte: Próprios autores, 2022.

O perfil de gerente é a coluna dorsal do SGCOE, neste perfil o gerente pode realizar o cadastro de membros da equipe, atribuir tarefas, monitorar tarefas, efetuar cadastro de cliente e exportar relatório em arquivo txt. para o computador.

##### 4.4.8.1 Cadastrar colaborador

Nesta opção o gerente pode cadastrar um perfil de colaborador. Para realizar o cadastrar o gerente deverá preencher todas a informações necessárias que o sistema exige, nome, cargo, e-mail, login e senha.

Figura 33: Formulário de cadastro de colaborador.

```

-FORMULÁRIO DE CADASTRO-
<<Após preencher um campo, pressione <ENTER> para seguir para o próximo>>

Nome completo: Luiz Carlos Machado
Cargo: Analista de sistemas
Email: Luiz@email.com
Cadastrar login: LuizLogin
Cadastrar senha: LuizSenha

CADASTRO REALIZADO COM SUCESSO!

Deseja continuar cadastrando?
<<Pressione <1> para continuar ou qualquer outra tecla para sair>>

```

Fonte: Próprios autores, 2022.

#### 4.4.8.2 Atribuir tarefas

Nesta opção o gerente pode atribuir tarefas para cada membro cadastrado no sistema, para atribuir uma tarefa o sistema pede primeiramente que identifique o colaborador através do ID. Após o colaborador ser identificado, o sistema vai exibir a seguinte tela.

Figura 34: Tela atribuir tarefas.

```

-ATRIBUIR TAREFAS(S)-
<<Após preencher o campo, pressione <ENTER> para seguir para o próximo campo>>

Dados do colaborador
ID: 7371
Nome: Luiz Carlos Machado
Cargo: Analista de sistemas
Email: Luiz@email.com

Tarefa(s)
Tarefas atribuídas <0>
Tarefas pendente(s) <0>
Tarefas concluída(s) <0>

Atribuindo Tarefa(s)
Titulo da tarefa:

```

Fonte: Próprios autores, 2022.

Nesse momento o sistema apresenta o perfil do colaborador com algumas informações básicas, como nome, e-mail e cargo, logo abaixo é exibido informações importantes sobre tarefas como, total de tarefas atribuídas, pendentes e concluídas. Na última linha o programa já pede que o gerente defina um título para a tarefa a ser atribuída e logo após a descrição da tarefa, como no exemplo abaixo.

Figura 35: Atribuindo tarefa.

The screenshot shows a software interface for assigning tasks. At the top, it says "-ATRIBUIR TAREFAS(S)-" and "AFTER preencher o campo, pressione <ENTER> para seguir para o próximo campo". The interface is divided into sections:

- Dados do colaborador**: Shows ID: 7371, Nome: Luiz Carlos Machado, Cargo: Analista de sistemas, Email: Luiz@email.com.
- Tarefa(s)**: Shows Tarefas atribuídas <0>, Tarefas pendente(s) <0>, and Tarefas concluída(s) <0>.
- Atribuindo Tarefa(s)**: Shows Titulo da tarefa: Escopo do projeto 'X' and Descrição da tarefa: Luiz, preciso do escopo 'X' pronto o mais rápido possível.

Fonte: Próprios autores, 2022.

Em seguida, o sistema pede que defina uma prioridade no qual o gerente precisa que o colaborador entenda de como deve ser tratada esta tarefa. Existem 3 níveis, o nível 1 é tratado pelo sistema como tarefa de alta prioridade, quando o colaborador identificar essa tarefa ele deverá tratar com urgência, o nível 2 é o nível mediano, normalmente tarefas em que o colaborador deverá se atentar com o prazo definido pelo gerente e por fim tarefas de nível 3, onde se trata de tarefas mais simples e normalmente com um prazo maior para concluir.

Figura 36: Nível de prioridade.

The screenshot shows a software interface for defining task priority levels. It starts with a header "Defina o nível de prioridade desta tarefa" and three options:

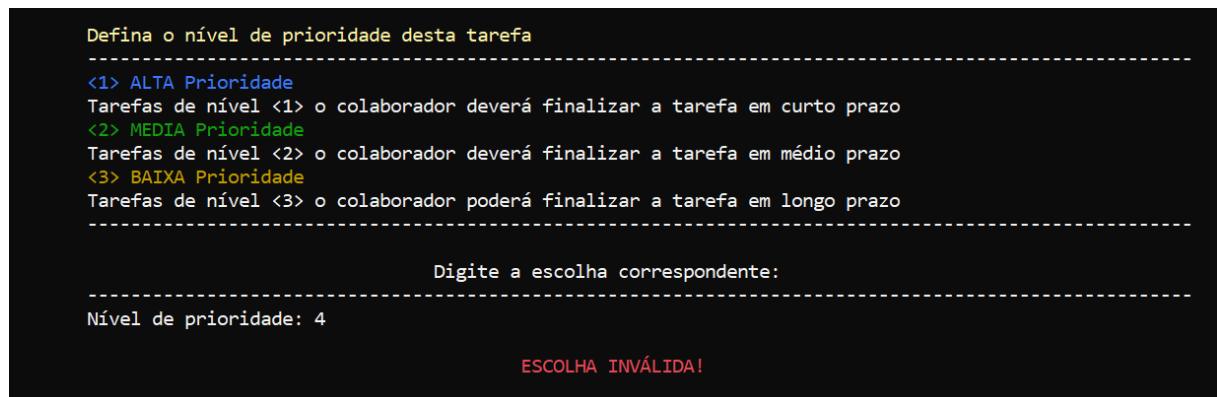
- <1> ALTA Prioridade: Tarefas de nível <1> o colaborador deverá finalizar a tarefa em curto prazo.
- <2> MEDIA Prioridade: Tarefas de nível <2> o colaborador deverá finalizar a tarefa em médio prazo.
- <3> BAIXA Prioridade: Tarefas de nível <3> o colaborador poderá finalizar a tarefa em longo prazo.

Below these options, it asks "Digite a escolha correspondente:" and shows "Nível de prioridade: 1".

Fonte: Próprios autores, 2022.

O sistema possui uma tratativa caso o usuário entrar com uma escolha indevida.

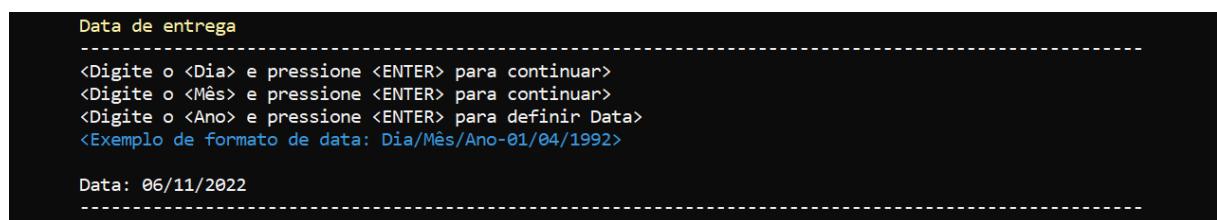
Figura 37: Nível de prioridade escolha inválida.



Fonte: Próprios autores, 2022.

Após definir um nível de prioridade, o programa irá pedir que o gerente defina um prazo para esta finalizar esta tarefa.

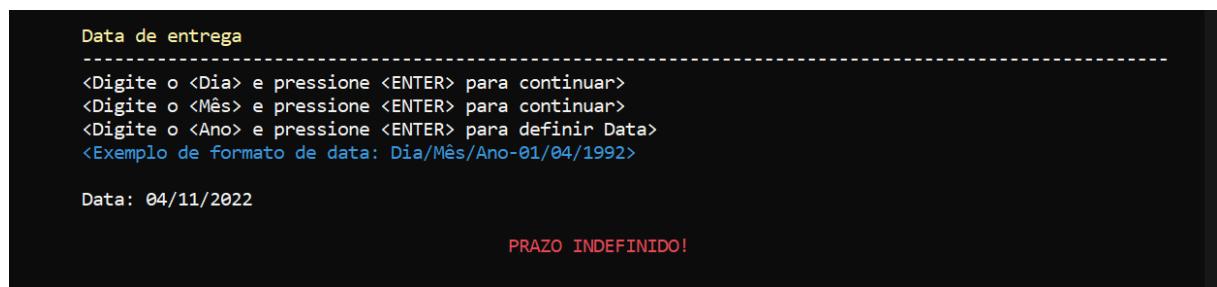
Figura 38: Data de entrega.



Fonte: Próprios autores, 2022.

O programa utiliza a data interna do Windows, aceitando apenas prazos reais, bloqueando qualquer informação incoerente.

Figura 39: Data de entrega prazo indefinido.



Fonte: Próprios autores, 2022.

Após preencher todos os campos corretamente a tarefas é atribuída com sucesso.

Figura 40: Tarefa atribuída com sucesso.

```

-ATTRIBUIR TAREFAS(S)-
<<Após preencher o campo, pressione <ENTER> para seguir para o próximo campo>>

Dados do colaborador
-----
ID: 7371
Nome: Luiz Carlos Machado
Cargo: Analista de sistemas
Email: Luiz@email.com

Tarefa(s)
-----
Tarefas atribuídas <0>
Tarefas pendente(s) <0>
Tarefas concluída(s) <0>

Atribuindo Tarefa(s)
-----
Título da tarefa: Escopo do projeto 'X'
-----
Descrição da tarefa: Luiz, preciso do escopo do projeto 'X' pronto o mais rápido possível.

Defina o nível de prioridade desta tarefa
-----
<1> ALTA Prioridade
Tarefas de nível <1> o colaborador deverá finalizar a tarefa em curto prazo
<2> MEDIA Prioridade
Tarefas de nível <2> o colaborador deverá finalizar a tarefa em médio prazo
<3> BAIXA Prioridade
Tarefas de nível <3> o colaborador poderá finalizar a tarefa em longo prazo

Digite a escolha correspondente:
-----
Nível de prioridade: 1

Data de entrega
-----
<Digite o <Dia> e pressione <ENTER> para continuar>
<Digite o <Mês> e pressione <ENTER> para continuar>
<Digite o <Ano> e pressione <ENTER> para definir Data>
<Exemplo de formato de data: Dia/Mês/Ano-01/04/1992>

Data: 06/11/2022

<TAREFA ATRIBUIDA COM SUCESSO!
Data de entrega definida: 6/11/2022, aproximadamente 1 dias corridos

<<Pressione 1 para continuar ou pressione qualquer outra tecla para voltar>>

```

Fonte: Próprios autores, 2022.

O sistema suporta no máximo 300 tarefas atribuídas para cada colaborador cadastrado.

#### 4.4.8.3 Monitorar tarefas

Nesta opção o gerente pode monitorar todas as tarefas atribuídas de cada colaborador, podendo monitorar tarefas concluídas, pendentes, excluir e até escrever pequenos comentários sobre a tarefa no qual o colaborador poderá visualizar em seu perfil.

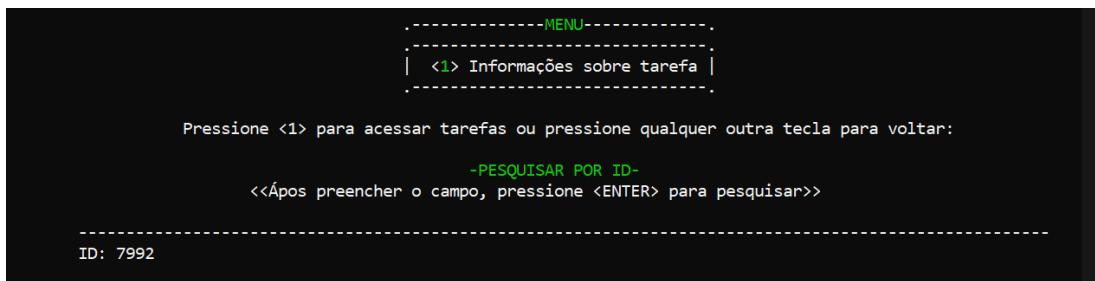
Figura 41: Tela monitorar tarefas.



Fonte: Próprios autores, 2022.

Para acessar essas funções mencionadas anteriormente, o gerente precisa escolher a opção 1 e digitar o ID do colaborador para abrir o leque de tarefas.

Figura 42: Menu monitorar tarefas.



Fonte: Próprios autores, 2022.

Figura 43: Monitorando tarefas.

```
-MONITORAR TAREFA(S)-

Dados do colaborador
-----
ID: 7992
Nome: Kaio Henrique Alves
Cargo: Programador Front-end
Email: kaio@email.com

Tarefa(s)
-----
Tarefas atribuídas <2>
Tarefas pendente(s) <2>
Tarefas concluída(s) <0>

TAREFA ID:1
Nível de prioridade: <MEDIA>
Título da tarefa: Tela Inicial
Descrição da tarefa: Kaio, preciso do conceito de arte da tela inicial do projeto 'X'
Data de entrega: 8/11/2022, aproximadamente 3 dias corridos

Status: <PENDENTE>

TAREFA ID:2
Nível de prioridade: <ALTA>
Título da tarefa: Tela Login e senha
Descrição da tarefa: Kaio, preciso que adicione um botão 'Sair', na tela de login e senha, por favor.
Data de entrega: 11/11/2022, aproximadamente 6 dias corridos

Status: <PENDENTE>

-----MENU-----
| <1> Add comentário
| <2> Excluir tarefa
| <3> Voltar

Digite a escolha correspondente:
```

Fonte: Próprios autores, 2022.

Após abrir o leque de tarefas do colaborador identificado, um menu com 3 opções é exibido logo em baixo da última tarefa apresentada, neste menu o gerente poderá adicionar comentário, excluir tarefa e voltar para o menu principal.

Para adicionar um comentário o gerente deverá escolher a opção 1 e em seguida o programa pedirá o ID da tarefa correspondente, esse ID é gerado automaticamente pelo algoritmo no momento que a tarefa é atribuída, respeitando a ordem de entrada das tarefas, como mostra a imagem a seguir.

Figura 44: Adicionando comentário.

```

Dados do colaborador
-----
ID: 7992
Nome: Kaio Henrique Alves
Cargo: Programador Front-end
Email: kaio@email.com

Tarefa(s)
-----
Tarefas atribuídas <2>
Tarefas pendente(s) <2>
Tarefas concluída(s) <0>

TAREFA ID:1
Nível de prioridade: <MEDIA>
Título da tarefa: Tela Inicial
Descrição da tarefa: Kaio, preciso do conceito de arte da tela inicial do projeto 'X'
Data de entrega: 8/11/2022, aproximadamente 3 dias corridos

-----
Status: <PENDENTE>

-----
TAREFA ID:2
Nível de prioridade: <ALTA>
Título da tarefa: Tela Login e senha
Descrição da tarefa: Kaio, preciso que adicione um botão 'Sair', na tela de login e senha, por favor.
Data de entrega: 11/11/2022, aproximadamente 6 dias corridos

-----
Status: <PENDENTE>

-----
.----- MENU -----.
| <1> Add comentário
| <2> Excluir tarefa
| <3> Voltar
.-----.

Digite a escolha correspondente:
-PESQUISAR POR ID DA TAREFA-
<<Após preencher o campo, pressione <ENTER> para pesquisar>>

-----
TAREFA ID: 2
Comentário: Kaio, por favor se atentar ao prazo, ok.

COMENTÁRIO FEITO COM SUCESSO!

Pressione qualquer tecla para continuar. . .

```

Fonte: Próprios autores, 2022.

Para excluir uma tarefa, o gerente deverá escolher a opção 2 e seguir os mesmos passos descritos anteriormente. Vale lembrar que o sistema mostra uma mensagem que a exclusão de uma tarefa é permanente.

Figura 45: Excluindo tarefa.

```

-MONITORAR TAREFA(S)-

Dados do colaborador
-----
ID: 7992
Nome: Kaio Henrique Alves
Cargo: Programador Front-end
Email: kaio@email.com

Tarefa(s)
-----
Tarefas atribuídas <2>
Tarefas pendente(s) <1>
Tarefas concluída(s) <1>

TAREFA ID:1
Nível de prioridade: <MEDIA>
Título da tarefa: Tela Inicial
Descrição da tarefa: Kaio, preciso do conceito de arte da tela inicial do projeto 'X'
Data de entrega: 8/11/2022, aproximadamente 3 dias corridos

Status: <PENDENTE>

TAREFA ID:2
Nível de prioridade: <ALTA>
Título da tarefa: Tela Login e senha
Descrição da tarefa: Kaio, preciso que adicione um botão 'Sair', na tela de login e senha, por favor.
Data de entrega: 11/11/2022, aproximadamente 6 dias corridos

Status: <CONCLUÍDO> em: 5/11/2022 às 13:42:59

Comentário:
-----
Comentário do gerente: <Obrigado Kaio!>

----- MENU -----
| <1> Add comentário
| <2> Excluir tarefa
| <3> Voltar
----- 

Digite a escolha correspondente:
-----
Digite o ID da tarefa:2

<<ATENÇÃO ESTA TAREFA SERÁ EXCLUÍDA PERMANENTEMENTE DO SISTEMA>>
<<Pressione <1> para confirmar ou pressione qualquer outra tecla para voltar>>

TAREFA EXCLUÍDA COM SUCESSO!
Pressione qualquer tecla para continuar. . .

```

Fonte: Próprios autores, 2022.

Após a exclusão a tarefa é removida de todos os perfis do sistema como mostra o exemplo a seguir.

Figura 46: Tela monitorar tarefas atualizado.

```

-DADOS DO COLABORADOR-
ID: 7992
Nome: Kaio Henrique Alves
Cargo: Programador Front-end
Email: kaio@email.com

-TAREFA(S)-
Tarefas atribuídas <1>
Tarefas pendente(s) <1>
Tarefas concluída(s) <0>

-TAREFA ID:1-
Nível de prioridade: <MEDIA>
Título da tarefa: Tela Inicial
Descrição da tarefa: Kaio, preciso do conceito de arte da tela inicial do projeto 'X'
Data de entrega: 8/11/2022, aproximadamente 3 dias corridos

>Status: <PENDENTE>

-MENU-
| <1> Add comentário
| <2> Excluir tarefa
| <3> Voltar

Digite a escolha correspondente:

```

Fonte: Próprios autores, 2022.

#### 4.4.8.4 Cliente

Na opção cliente o gerente tem acesso ao menu contendo o sistema de cadastro de cliente, pesquisa e lista de cadastro, um sistema simples, no qual o algoritmo foi explicado anteriormente.

Figura 47: Menu cliente.

```

-PERFIL LOGADO: <<GERENTE>> <<CLIENTE>>

-MENU-
| <1> Cadastrar cliente
| <2> Pesquisar
| <3> Lista de clientes cadastrado(s)
| <4> Voltar

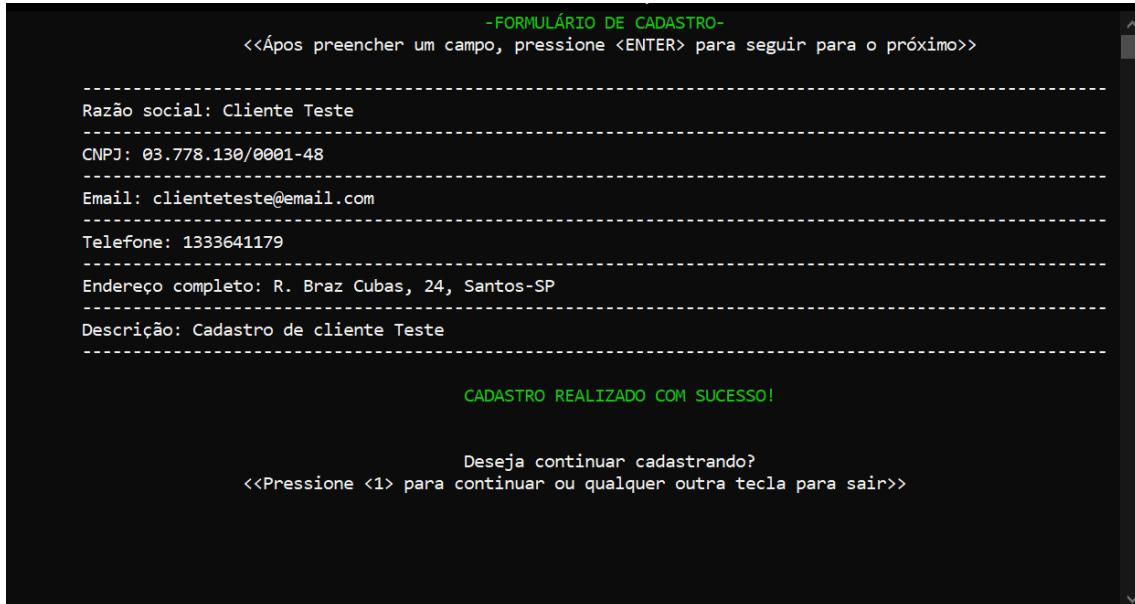
Digite a escolha correspondente:

```

Fonte: Próprios autores, 2022.

Formulário de cadastro de cliente possui os seguintes campos, razão social, CNPJ, E-mail, telefone, endereço e descrição.

Figura 48: Formulário de cadastro de cliente.



Fonte: Próprios autores, 2022.

O sistema comporta no máximo 300 clientes cadastrados.

Figura 49: Lista de cadastro de cliente.

<b>-LISTA DE CADASTRO(S)-</b>	
Razão social:	Cliente Teste
CNPJ:	03.778.130/0001-48
Telefone:	1333641179
Email:	clienteteste@email.com
Endereço:	R. Braz Cubas, 24, Santos-SP
Descrição:	Cadastro de cliente Teste
Razão social:	Cliente Teste 2
CNPJ:	04.779.131/0001-48
Telefone:	1333642425
Email:	clienteteste2@email.com
Endereço:	Rua Francisco da Costa Pina, 182
Descrição:	Vila Thomaz Coelho
Pressione qualquer tecla para continuar. . .	

Fonte: Próprios autores, 2022.

#### 4.4.8.5 Exportar relatório

Nesta opção o sistema copila todos os dados dos colaboradores e suas respectivas tarefas, exportado como arquivo.txt para a pasta em que o SGCOE está alocado no computador.

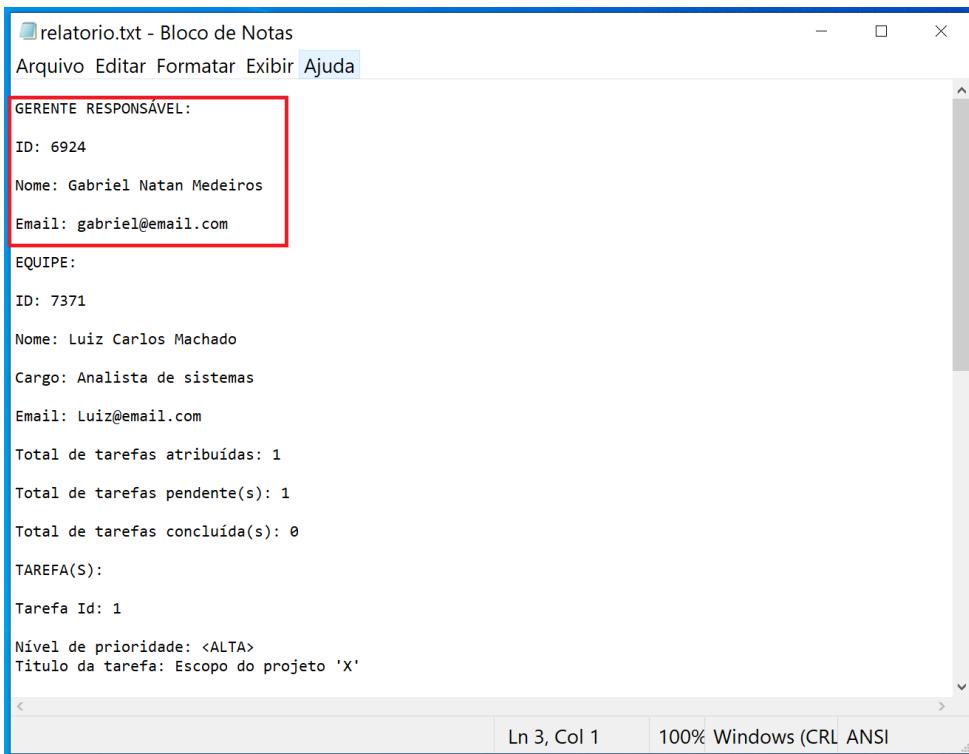
Figura 50: Exportando relatório.

The screenshot shows a terminal window with a black background and white text. It displays the export of a report from the SGCOE system. The report includes information about a responsible manager, their team, tasks assigned, pending tasks, completed tasks, and details of a specific task. At the bottom, it shows a success message and a prompt to press any key to continue.

```
GERENTE RESPONSÁVEL:  
ID: 25636  
Nome: Gabriel Nathan Medeiros  
Email: Gabriel@email  
EQUIPE:  
ID: 25786  
Nome: Luiz Carlos Machado  
Cargo: Analista de Sistemas  
Email: luiz@email.com  
Total de tarefas atribuídas: 1  
Total de tarefas pendente(s): 1  
Total de tarefas concluída(s): 0  
TAREFA(S):  
Tarefa Id: 1  
Nível de prioridade: <ALTA>  
Título da tarefa: Apresentar escopo do projeto  
Descrição da tarefa: Luiz, preciso que apresente o escopo do projeto de rede para o cliente  
Data limite para entrega: 25/11/2022  
Status: <PENDENTE>  
  
RELATÓRIO EXPORTADO COM SUCESSO!  
Pressione qualquer tecla para continuar. . .
```

Fonte: Próprios autores, 2022.

Figura 51: Arquivo txt.



The screenshot shows a Windows Notepad window with the title bar 'relatorio.txt - Bloco de Notas'. The menu bar includes 'Arquivo', 'Editar', 'Formatar', 'Exibir', and 'Ajuda'. The main content area contains the following text:

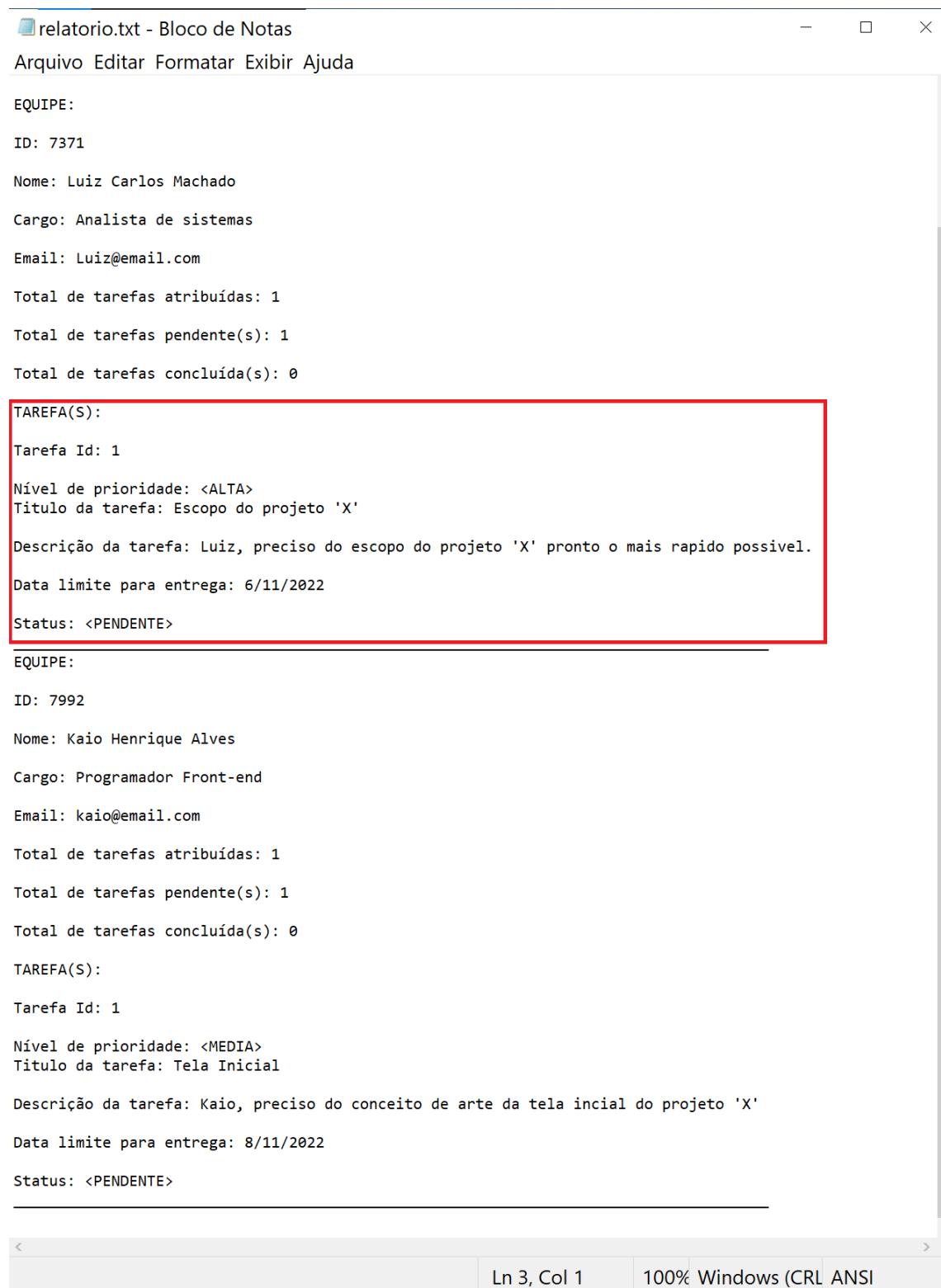
```
GERENTE RESPONSÁVEL:  
ID: 6924  
Nome: Gabriel Natan Medeiros  
Email: gabriel@email.com  
  
EQUIPE:  
ID: 7371  
Nome: Luiz Carlos Machado  
Cargo: Analista de sistemas  
Email: Luiz@email.com  
Total de tarefas atribuídas: 1  
Total de tarefas pendente(s): 1  
Total de tarefas concluída(s): 0  
  
TAREFA(S):  
Tarefa Id: 1  
Nível de prioridade: <ALTA>  
Titulo da tarefa: Escopo do projeto 'X'
```

The first three lines under 'GERENTE RESPONSÁVEL:' are highlighted with a red rectangular box.

Fonte: Próprios autores, 2022.

No arquivo inicialmente vai apontar os dados cadastrais do gerente responsável pela equipe, logo abaixo mostra os dados cadastrais de cada colaborador e o status de cada tarefa atribuída.

Figura 52: Relatório Arquivo txt.



The screenshot shows a Microsoft Notepad window with two distinct sections of text, each representing a team report. The first section is for 'EQUIPE:' ID 7371, and the second is for 'EQUIPE:' ID 7992. Both sections include details about team members, assigned tasks, pending tasks, and completed tasks. A red box highlights the task details for the first team. The Notepad window has standard Windows-style controls at the top and a status bar at the bottom indicating the current file path and encoding.

**EQUIPE:**  
ID: 7371  
Nome: Luiz Carlos Machado  
Cargo: Analista de sistemas  
Email: Luiz@email.com  
Total de tarefas atribuídas: 1  
Total de tarefas pendente(s): 1  
Total de tarefas concluída(s): 0

**TAREFA(S):**

Tarefa Id: 1  
Nível de prioridade: <ALTA>  
Título da tarefa: Escopo do projeto 'X'  
Descrição da tarefa: Luiz, preciso do escopo do projeto 'X' pronto o mais rápido possível.  
Data limite para entrega: 6/11/2022  
Status: <PENDENTE>

**EQUIPE:**  
ID: 7992  
Nome: Kaio Henrique Alves  
Cargo: Programador Front-end  
Email: kaio@email.com  
Total de tarefas atribuídas: 1  
Total de tarefas pendente(s): 1  
Total de tarefas concluída(s): 0

**TAREFA(S):**

Tarefa Id: 1  
Nível de prioridade: <MEDIA>  
Título da tarefa: Tela Inicial  
Descrição da tarefa: Kaio, preciso do conceito de arte da tela inicial do projeto 'X'  
Data limite para entrega: 8/11/2022  
Status: <PENDENTE>

< >  
Ln 3, Col 1 | 100% Windows (CRL ANSI)

Fonte: Próprios autores, 2022.

#### 4.4.8.6 Criptografia

##### 4.4.8.6.1 O que é criptografia de dados?

Segundo (Totvs, 2022) a criptografia é um conjunto de técnicas pensadas para proteger uma informação de modo que apenas o emissor e receptor consigam compreendê-la. (Totvs, 2022) explica que, em geral, são usados algoritmos para realizar a codificação e para decodificação é necessário ter acesso à chave utilizada no primeiro processo, ou seja, o princípio básico da criptografia é permitir que duas pessoas compartilham secretamente mensagens entre si, sem que elas sejam acessadas por terceiros.

(Ciriaco, 2015) comenta que o protocolo de criptografia existem desde a antiguidade, com o primeiro sistema de criptografia conhecido tendo surgido no Egito, cerca de 1.900 anos antes de Cristo. Segundo o autor, a criptografia tem um apelo especial para assuntos ligados à guerra, mas comerciantes e governantes também podem ver uma saída na criptografia para evitar que pessoas não autorizadas descubram informações sobre suas estratégias, por exemplo. A ideia básica é que este sistema de técnicas cifice uma informação que será somente decifrada por pessoas autorizadas, sem acessos indevidos no caminho.

##### 4.4.8.6.2 Quais são os tipos de criptografia?

###### 4.4.8.6.2.1 Criptografia simétrica

Segundo (Totvs, 2022) a criptografia simétrica é o tipo mais tradicional e provavelmente o sistema que as pessoas estão mais familiarizadas. Neste tipo, a criptografia é realizada com base em uma única chave utilizada para criptografar e também descriptografar uma mensagem. De acordo com (Totvs, 2022), sua principal aplicação é na proteção de dados em repouso, como em bancos de dados ou discos rígidos, isso porque é necessário contar com um canal seguro para transmitir a mensagem.

(Totvs, 2022) comenta que entre seus principais benefícios, a criptografia se destaca por ser mais rápida e por ser ideal para proteger dados que vão ficar em único

local, entretanto como desvantagem, destaca-se a dificuldade de distribuição segura de chaves.

#### 4.4.8.6.2.2 Criptografia Assimétrica

De acordo com (Kovacs, 2021), a criptografia assimétrica ou de chave pública usa um par de chaves. O autor comenta que esse nível adicional de segurança aumenta a proteção dos dados. Nesse caso, cada chave tem um único propósito.

(Kovacs, 2021) explica que existe uma chave pública que pode ser trocada com qualquer pessoa, em qualquer rede. Essa chave contém as informações sobre como criptografar os dados e todos podem usá-la. Mas também existe uma chave privada. Essa chave privada não é compartilhada e contém as informações sobre como descriptografar a mensagem. Segundo o autor, ambas as chaves são geradas por um algoritmo que usa grandes números primos para criar duas chaves únicas, matematicamente vinculadas.

#### 4.4.8.6.2.3 Funções de Hash

Segundo (Kovacs, 2021) as funções de hash são outra maneira pela qual a criptografia pode proteger as informações. Ao invés de chaves, depende de algoritmos para transformar qualquer entrada de dados em uma sequência de caracteres de comprimento fixo. Segundo o autor, as funções de hash também diferem de outras formas de criptografia porque funcionam apenas de uma maneira, ou seja, significa que não se pode transformar um hash em seus dados originais.

De acordo com (Kovacs, 2021), Hashes são essenciais para o gerenciamento de blockchain porque podem criptografar grandes quantidades de informações sem comprometer os dados originais. Hashes podem agir como impressões digitais para quaisquer dados que foram criptografados.

#### 4.4.8.6.2.4 Assinaturas digitais

Para (Kovacs, 2021), uma assinatura digital é fundamental para garantir a segurança, autenticidade e integridade dos dados em uma mensagem, software ou documento digital, agindo de forma semelhante às assinaturas físicas e são uma maneira única de vincular sua identidade aos dados, portanto, agem como uma forma de verificar as informações. Entretanto, em vez de representar sua identidade como assinaturas físicas, as assinaturas digitais são baseadas na criptografia de chaves públicas.

Segundo (Kovacs, 2021) a assinatura digital vem como código, anexado aos dados graças às duas chaves de autenticação mútua. O remetente cria a assinatura digital usando uma chave privada para criptografar os dados relacionados à assinatura, com o receptor obtendo a chave pública do assinante para descriptografar os dados.

#### 4.4.8.7 Criptografia SGCOE

Para encriptar o arquivo relatório.txt exportado para o computador utilizamos o método simétrica que consiste em usar a mesma chave na codificação e na descodificação. No algoritmo desenvolvido em C usamos uma função simples que utiliza a tabela ASCII para codificar as informações gravadas no arquivo.

Figura 53: Bloco de códigos 1 da função criptografia.

```

54 void criptografia(int opcao)
55 {
56     FILE *origem, *destino;
57
58     setlocale (LC_ALL, "Portuguese");
59
60     if (opcao == 1)
61     {
62         system("cls");
63         textColor(10);
64         printf("\t\t\t\t\t\t-RELATÓRIO-\n\n");
65         textColor(15);
66
67         origem = fopen("relatorio.txt", "r");
68         destino = fopen("relatorioCript.txt", "w");
69
70 }
```

Fonte: Próprios autores, 2022.

Primeiro criamos a função do tipo **void**, que recebe como parâmetro um número inteiro, esse numero poder ser 1 ou 2. Para criptografar um arquivo na hora de chamar a função passamos como parâmetro o número 1, e para descriptografar o número 2.

Seguindo a sequência foi criado 2 ponteiros do tipo **FILE**, origem e destino, origem vai receber o arquivo exportado chamado **relatorio.txt** contendo todos os dados compilados e destino vai criar o mesmo arquivo porem criptografado com o nome de **relatorioCript.txt**.

Figura 54: Bloco de códigos 2 da função criptografia.

```

70     if(origem != NULL && destino != NULL)
71     {
72         char c;
73         int contador = 0;
74         int i = 0;
75
76         while(( c = fgetc(origem)) != EOF)
77         {
78             contador++;
79             printf("%c", c);
80             i = c;
81             i = i * contador;
82             fprintf(destino,"%i ", i);
83         }
84         fclose(origem);
85         fclose(destino);
86         remove("relatorio.txt");
87     }

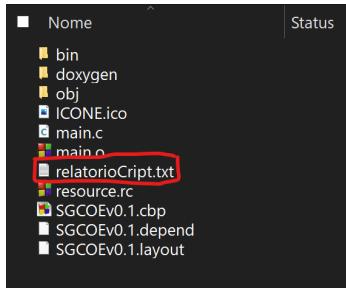
```

Fonte: Próprios autores, 2022.

No **if** verificar se os ponteiros diferem de nulo, se verdadeiro executa o laço de repetição **while** enquanto o **fgetc** não retornar o fim do arquivo, a cada repetição a variável contadora irá somar +1, printa na tela cada caractere do arquivo de origem e em seguida a variável “i” vai receber o código ASCII do caractere armazenado na variável **c**, seguindo a sequência, a operação matemática(**i = i \* contador**) irá criptografar o caractere.

Por exemplo, o caractere “A” maiúsculo na tabela ASCII representa o número decimal 65. Supondo que a variável “c” possui armazenado o caractere “A” logo quando  $i = c$ , então  $i = 65$ . No primeiro laço de repetição a variável contadora que entra com o valor de 0 irá somar +1, quando  $(i * \text{contador})$  é o mesmo que  $(65 * 1)$  o resultado irá atribuir variável “i” novamente, em seguida o comando **fprintf** vai gravar no arquivo o resultado da operação. No segundo laço de repetição irá seguir a mesma lógica usando o próximo caractere do arquivo, entre tanto o valor de variável contadora agora é ser 2. O algoritmo vai se repetir até o fim do arquivo, dessa maneira os dados do arquivo são criptografados. Quando o **while** terminar o comando **fclose** fecha os arquivos origem e destino, porem o arquivo de origem **relatorio.txt** é removido da pasta havendo apenas existência do arquivo **relatorioCript.txt**.

Figura 55: Bloco de códigos 3 da função criptografia.



Fonte: Próprios autores, 2022.

Figura 56: Bloco de códigos 3 da função criptografia.

```
21380 28758 24725 12528 6944 10682 2190 2208 18564 24642 25868
21728 24300 2732 22709 23028 7328 26688 22487 26448 23533 23868
22795 27140 7584 26656 24139 26408 24108 24442 26738 28304 24745
9840 28405 10168 14442 8000 12299 2520 2530 21336 28305 29696
24929 27864 8288 26000 26361 8384 30508 25608 30218 26866 27234
25996 30935 8648 26829 30192 30830 27126 29708 32292 -5263
27800 27063 11200 32315 11562 16414 9888 13680 2860 2870 24192
18783 23780 20078 20440 19045 11768 24485 12136 17228 2980
2994 25280 29197 34424 30683 31088 29585 9792 22411 30800
17922 9920 15239 3120 3130 24492 -5985 37288 32017 34344
10208 32000 32421 10304 36176 36936 34125 36186 37278 34440
32908 32010 33108 33532 19314 10688 20108 21840 25612
28392 22035 21080 3410 28728 36015 39908 40365 37368 38517
11136 34900 33950 11232 48392 34241 40356 35855 36312
34629 28764 11488 23400 40432 41268 36663 41868 36865 40260
42572 35696 42066 11840 37471 42780 36927 41514 42000 41736
12064 37800 42069 12160 42672 43548 42513 40704 38885 44776
42957 3880 3890 26520 39491 45080 38907 44916 41475 -9900
-11513 44178 12768 40000 38897 12864 46748 39188 46170 41006
41514 39576 23722 13120 31236 48204 43365 50508 18260 13312
46708 47652 42319 41580 44285 48530 46953 13568 48025 49842
43127 13694 41613 48168 49134 43632 49793 43834 47854 50576
44137 14016 48729 14080 44541 50830 43857 49284 49840 49506
14304 44800 49839 14400 50512 51528 50283 48124 45955 52896
50727 14656 45906 46468 14752 52668 46763 46400 46965 14912
52308 45396 53466 45598 15872 52392 15136 46926 51308 49980
48177 52580 55564 48480 4810 4820 32844 46948 56260 47142
15584 52704 51345 53418 51555 57072 49793 15808 55448 48112
56658 48306 15968 50500 55118 58232 57342 50904 52015 49882
29406 16256 25456 27030 24017 25088 25137 24158 25750 24768
25850 25900 5190 5200 43243 60552 50731 60784 61425 60490
30566 16896 31740 42400 36639 41496 36244 36846 41730 45024
37053 33356 5390 51300 51395 51490 51585 51680 51775 51870
51965 52060 52155 52250 52345 52440 52535 52630 52725 52820
52915 53010 53105 53200 53295 53390 53485 53580 53675 53770
| 53865 53960 54055 54150 54245 54340 54435 54530 54625 54720 54815 54
```

Fonte: Próprios autores, 2022.

Apenas um gerente cadastrado possui a chave para descriptografar o arquivo, pois o próprio sistema tem essa função que está disponível no menu de opções do gerente, acessado usando uma senha e login destinado a ele. Ao escolher essa opção, o algoritmo importa da pasta de origem o arquivo (relatorioCript.txt) e descifra usando o mesmo método para criptografar mais aplicado de maneira inversa.

Figura 57: Bloco de códigos 3 da função criptografia.

```
88        else
89        {
90            puts ("Erro de leitura/escrita nos arquivos..."); 
91        }
92    }
93    else if(opcao == 2) // Se for descriptografia
94    {
95        system("cls");
96        origem = fopen("relatorioCript.txt", "r");
97        destino = fopen("relatorioDescript.txt", "w");
98
99        if(origem != NULL && destino != NULL)
100        {
101            int contador = 0;
102            char c[100];
103            int i = 0;
104        }

```

Fonte: Próprios autores, 2022.

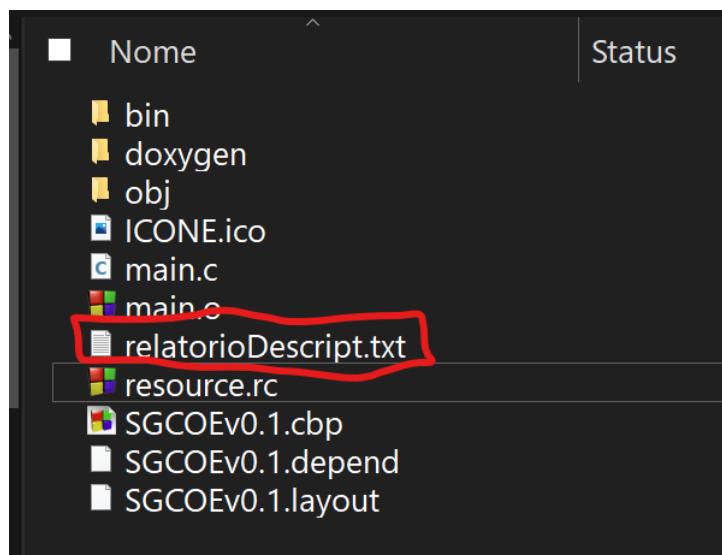
Figura 58: Bloco de códigos 4 da função criptografia.

```
105        while(fscanf(origem, "%s", c) != EOF)
106        {
107            contador++;
108            i = atoi(c);
109            i /= contador;
110            printf ("%c", i);
111            fprintf(destino, "%c", i);
112        }
113        fclose(origem);
114        fclose(destino);
115        remove ("relatorioCript.txt");
116        textColor(10);
117        printf ("\n\t\t\t\t\t\tRELATÓRIO DESCRIPTOGRAFADO COM SUCESSO!\n");
118        textColor(15);
119        printf ("\n\t\t\t\t\t\t\t\t");
120        system("pause");
121    }
```

Fonte: Próprios autores, 2022.



Figura 61: Arquivo relatorioDescript.txt.



Fonte: Próprios autores, 2022.

Figura 62: Arquivo descriptografado.

```
*relatorioDescript.txt - Bloco de Notas
Arquivo Editar Formatar Exibir Ajuda
GERENTE RESPONSÁVEL:
ID: 25636
Nome: Gabriel Nathan Medeiros
Email: Gabriel@email
EQUIPE:
ID: 25786
Nome: Luiz Carlos Machado
Cargo: Analista de Sistemas
Email: luiz@email.com
Total de tarefas atribuídas: 1
Total de tarefas pendente(s): 1
Total de tarefas concluída(s): 0
TAREFA(S):
Tarefa Id: 1
Nível de prioridade: <ALTA>
Título da tarefa: Apresentar escopo do projeto
Descrição da tarefa: Luiz, preciso que apresente o escopo do projeto
Data limite para entrega: 25/11/2022
Status: <PENDENTE>
```

Ln 1, Col 1 100% Windows (CRL ANSI)

Fonte: Próprios autores, 2022.

#### 4.4.9 Perfil Colaborador

Cada colaborador tem acesso ao sistema através do login e senha cadastrados pelo gerente de projeto, cada acesso é individual, ou seja, o colaborador tem acesso apenas a suas tarefas, não sendo possível acessar informações de outros colaboradores ou gerente.

Figura 63: Tela login colaborador.

```
Tentativa:<1>
-----
Entre com o login: KaioLogin
Entre com a senha: *****
-----
>>LOGADO COM SUCESSO<<
Pressione qualquer tecla para continuar...
```

Fonte: Próprios autores, 2022.

Figura 64: Tela menu colaborador.

```
PERFIL LOGADO: <<Kaio Henrique Alves>>

-----MENU-----
| <> Tarefas Pendentes
| <> Tarefas Concluidas
| <> Voltar para tela de login
| <> Sair
-----
Digite a escolha correspondente:
```

Fonte: Próprios autores, 2022.

Na tela de menu de principal, o colaborador tem as seguintes opções, tarefas pendentes, tarefas concluídas, voltar e sair.

#### 4.4.9.1 Tarefas pendentes

Nesta opção o colaborador tem acesso a todas as tarefas atribuídas, como seus detalhes, ID, total de tarefas pendentes e concluídas, nível de prioridade, descrição e comentários adicionados pelo gerente, após visualizar todos os detalhes de suas tarefas, logo abaixo o colaborador encontrará uma opção onde ele pode marcar como concluída, após finalizar uma tarefa.

Figura 65: Tela tarefas colaborador.

```

-TAREFAS(S)-

Dados do colaborador
-----
ID: 7992
Nome: Kaio Henrique Alves
Cargo: Programador Front-end
Email: kaio@email.com

Tarefa(s)
-----
Tarefas atribuidas <2>
Tarefas pendente(s) <2>
Tarefas concluída(s) <0>

TAREFA ID:1
Nível de prioridade: <MEDIA>
Título da tarefa: Tela Inicial
Descrição da tarefa: Kaio, preciso do conceito de arte da tela inicial do projeto 'X'
Data de entrega: 8/11/2022, aproximadamente 3 dias corridos

TAREFA ID:2
Nível de prioridade: <ALTA>
Título da tarefa: Tela Login e senha
Descrição da tarefa: Kaio, preciso que adicione um botão 'Sair', na tela de login e senha, por favor.
Data de entrega: 11/11/2022, aproximadamente 6 dias corridos

Comentário:
-----
Comentário do gerente: <Kaio, por favor se atentar ao prazo, ok.>

Pressione qualquer tecla para continuar. . .
----- MENU -----
| <1> Finalizar tarefa |
----- . .

Pressione <1> para finalizar alguma tarefa ou pressione qualquer outra tecla para voltar:_

```

Fonte: Próprios autores, 2022.

Ao escolher a opção de concluir uma tarefa o sistema irá pedir que o usuário entre com o ID da tarefa, após digitar o ID o programa marcará a tarefa como

“Concluída” com a data e hora do momento exato que a tarefa foi finalizada, logo a tarefa é removida da tela de tarefas pendentes e aparecerá na opção de tarefas concluídas.

Figura 66: Tela concluindo tarefa.

```

TAREFA ID:2
Nível de prioridade: <ALTA>
Título da tarefa: Tela Login e senha
Descrição da tarefa: Kaio, preciso que adicione um botão 'Sair', na tela de login e senha, por favor.
Data de entrega: 11/11/2022, aproximadamente 6 dias corridos

Comentário:
-----
Comentário do gerente: <Kaio, por favor se atentar ao prazo, ok.>

-----
Pressione qualquer tecla para continuar. . .

----- MENU -----
|   <1> Finalizar tarefa   |
-----

Pressione <1> para finalizar alguma tarefa ou pressione qualquer outra tecla para voltar:
-----
Digite o ID da tarefa: 2
-----
Tarefa, ID: 2, CONCLUÍDA COM SUCESSO!
-----
Pressione qualquer tecla para continuar. . .

```

Fonte: Próprios autores, 2022.

Figura 67: Tela tarefa concluída.

```

-TAREFAS(S) CONCLUÍDAS-

Dados do colaborador
-----
ID: 7992
Nome: Kaio Henrique Alves
Cargo: Programador Front-end
Email: kaio@email.com

Tarefa(s)
-----
Tarefas atribuídas <2>
Tarefas pendente(s) <1>
Tarefas concluída(s) <1>
-----

Tarefa(s)
-----
TAREFA ID:2
Nível de prioridade: <ALTA>
Título da tarefa: Tela Login e senha
Descrição da tarefa: Kaio, preciso que adicione um botão 'Sair', na tela de login e senha, por favor.
Data limite para entrega: 11/11/2022

Entregue em: 5/11/2022 ás 13:42:59

Comentário:
-----
Comentário do gerente: <Kaio, por favor se atentar ao prazo, ok.>

-----
Pressione qualquer tecla para continuar. . .

```

Fonte: Próprios autores, 2022.



## 5 CONSIDERAÇÕES FINAIS

A Devcoast foi contratada para projetar a estrutura de rede e desenvolver um gerenciador de tarefas em linguagem C para a startup Choveu? Onde encheu?(startup desenvolvida no primeiro PIM), com muita satisfação podemos afirmar conseguimos alcançar os objetivos propostos, com o conhecimento adquirido ao longo deste semestre. Apresentar e desenvolver o planejamento da estrutura de rede e o gerenciador de tarefas SGCOE foi bastante desafiador, entretanto muito gratificante, pois ao final entregamos um plano de infra estrutura de rede coeso e um programa funcional que atendeu os requisitos do cliente.

## REFERÊNCIAS

ESPINHA, Roberto Gil. **Gestão de Tarefas:** Tudo sobre como gerenciar atividades. Artia. 2022. Disponível em: <https://artia.com/gestao-de-tarefas/>. Acesso em: 12 de Novembro de 2022.

MACHADO, Luiz Carlos de Araújo. **CHOVEU? ONDE ENCHEU?:** Projeto de startup sobre alagamentos em São Vicente. UNIP. Santos. 2022. p. 34.

**Noleto, Cairo. Linguagem C: o que é e quais os principais fundamentos.** Betrybe. 2022. Disponível em: <https://blog.betrybe.com/linguagem-de-programacao/linguagem-c/>. Acesso em: 12 de Novembro de 2022.

PIRES, Raphael. 2018. **Gerenciador de tarefas:** o que é e quais suas funcionalidades. Rockcontent. 2018. Disponível em: <https://rockcontent.com/br/blog/o-que-e-gerenciador-de-tarefas/>. Acesso em: 12 de Novembro de 2022.

PROJ4ME, Redação. **Entenda o que é um Gerenciador de Tarefas e suas funcionalidades.** Proj4me. 2021. Disponível em: <https://proj4.me/blog/gerenciador-de-tarefas>. Acesso em: 12 de Novembro de 2022.

TOTVS, Equipe. **Criptografia:** tipos, exemplos e importância nas empresas. Totvs. 2022. Disponível em: <https://www.totvs.com/blog/negocios/criptografia/#:~:text=A%20criptografia%20%C3%A9%20um%20conjunto,mensagens%20ou%20em%20pagamentos%20online>. Acesso em: 20 de Novembro de 2022.

CIRIACO, Douglas. **O que é criptografia e por que você deveria usá-la.** Canaltech. 2015. Disponível em: <https://canaltech.com.br/seguranca/o-que-e-criptografia-e-por-que-voce-deveria-usa-la/>. Acesso em: 20 de Novembro de 2022.

KOVACS, Leandro. **O que é criptografia e quais os tipos?**. Tecnoblog. 2021. Disponível em: <https://tecnoblog.net/responde/o-que-e-criptografia-e-quais-os-tipos/>. Acesso em: 20 de Novembro de 2022.