

UNIVERSIDADE PAULISTA – UNIP EAD

Projeto Integrado Multidisciplinar

Curso Superior de Tecnologia em Análise e Desenvolvimento de Sistemas

GABRIEL SILVA RODRIGUES - R.A: 0411858

POLO: DUTRA (SÃO JOSÉ DOS CAMPOS) - TUTOR: VIVIANE

JOÁS ALMEIDA DOS SANTOS - R.A: 0448739

POLO: ITATIBA (ITATIBA) - TUTOR: PAULO

SCHNEIDER SIMILIEN - R.A: 0448458

POLO: JUNDIAI (JUNDIAI) - TUTOR: ANGEL ANTONIO

LARISSA DA COSTA CARVALHO - R.A: 0446897

POLO: ARTHUR NOGUEIRA (COSMÓPOLIS) - TUTOR: INGRID

APARECIDO ANGELO DA CRUZ - R.A: 0446304

POLO: VILA FORMOSA (SÃO PAULO) - TUTOR: CARLOS BATISTA

VINICIUS FERNANDES DA SILVA - R.A: 0448848

POLO: BRASILIA - TUTOR: VANESSA SANTOS LESSA

SISTEMA EM LINGUAGEM C - CADASTRO DE PACIENTES COM COVID-19

GABRIEL SILVA RODRIGUES - R.A: 0411858

JOÁS ALMEIDA DOS SANTOS - R.A: 0448739

SCHNEIDER SIMILIEN - R.A: 0448458

LARISSA DA COSTA CARVALHO - R.A: 0446897

APARECIDO ANGELO DA CRUZ - R.A: 0446304

VINICIUS FERNANDES DA SILVA - R.A: 0448848

PROJETO INTEGRADO MULTIDISCIPLINAR IV
SISTEMA EM LINGUAGEM C - CADASTRO DE PACIENTES COM COVID-19

Projeto Integrado Multidisciplinar IV
para obtenção do título de tecnólogo em
Análise e Desenvolvimento de Sistemas,
apresentado à Universidade Paulista – UNIP EaD.

Orientadores:

VIVIANE - POLO: DUTRA (SÃO JOSÉ DOS CAMPOS)

PAULO - POLO: ITATIBA (ITATIBA)

ANGEL ANTONIO - POLO: JUNDIAI (JUNDIAI)

INGRID - POLO: ARTHUR NOGUEIRA (COSMÓPOLIS)

CARLOS BATISTA - POLO: VILA FORMOSA (SÃO PAULO)

VANESSA SANTOS LESSA - POLO: BRASILIA

RESUMO

Em 31 de Dezembro de 2019 a OMS é alertada sobre um número crescente de casos de pneumonia na cidade de Wuhan, na China. Caracterizado de um novo coronavírus que ainda não havia sido catalogado até então. Alguns meses depois, em 11 de Fevereiro o novo vírus foi intitulado de SARS-CoV-2, vírus responsável por causar a doença COVID-19.

Em 11 de Março de 2020 a OMS caracterizou a COVID-19 como uma pandemia global, uma vez que o número de pessoas infectadas pelo novo vírus estava tomando alta crescente, se alastrando assim em vários países e regiões do mundo. Diante deste problema que a população está vivendo, a sociedade tem procurado soluções e inovações afins de amenizar os casos e também a propagação do novo coronavírus. Desta forma, este projeto possui como finalidade a execução da fundamentação teórica e a prática efetivando assim um completo processo de ensino e aprendizagem.

Os conceitos adquiridos são utilizados para apresentar um sistema em linguagem C, afim de auxiliar hospitais e unidades de pronto atendimento (UPA) a realizarem o cadastro de pacientes que possam ser diagnosticados com COVID-19 e que necessitam de um acompanhamento para que essa informação possa ser enviada para a Secretária de Saúde, por meio do CEP e idade do indivíduo.

Por fim, este projeto visa ajudar essas entidades a catalogarem a quantidade e também a distribuição dos casos de COVID-19 que vem sendo apresentados na região.

PALAVRAS CHAVES: Sistema de Cadastro de Pacientes, Sistema em Linguagem C.

ABSTRACT

As of December 31, 2019, WHO is alerted to an increasing number of pneumonia cases in Wuhan City, China. Characterized from a new coronavirus that had not yet been cataloged. A few months later, on February 11, the new virus was named SARS-CoV-2, the virus responsible for causing COVID-19 disease.

On March 11, 2020, WHO characterized COVID-19 as a global pandemic, as the number of people infected by the new virus was increasing, spreading in several countries and regions of the world.

Faced with this problem that the population is experiencing, society has been looking for solutions and innovations in order to alleviate the cases and also the spread of the new coronavirus. Thus, this project aims to carry out the theoretical foundation and practice, thus effecting a complete teaching and learning process.

The acquired concepts are used to present a system in C language, in order to help hospitals and emergency care units (UPA) to carry out the registration of patients who may be diagnosed with COVID-19 and who need follow-up so that this information can be sent to the Secretary of Health, through the zip code and age of the individual.

Finally, this project aims to help these entities to catalog the quantity and also the distribution of COVID-19 cases that have been presented in the region.

KEYWORDS: Patient Registration System, Language System C.

Sumário

1. INTRODUÇÃO.....	6
2. A LINGUAGEM C	6
3. METODOLOGIA USADA NO PROJETO	7
4. SOBRE O PROJETO (VARIÁVEIS E COMANDOS)	12
5. MANUAL DE INSTALAÇÃO.....	12
6. SISTEMA DE CADASTRO DE PACIENTES COM COVID-19	13
6.1 REGISTRO DO USUÁRIO.....	14
6.2 TELA DE LOGIN	15
6.3 MENU PRINCIPAL.....	16
6.4 CADASTRO DO PACIENTE	17
6.5 CONSULTA DE PACIENTE	18
6.6 DADOS DO SISTEMA.....	19
6.7 SAIR.....	20
7. COMPILAÇÃO DO PROGRAMA.....	20
8. CONCLUSÃO	21
9. REFERÊNCIAS	22

1. INTRODUÇÃO

Em 11 de Março de 2020 a OMS caracterizou a COVID-19 como uma pandemia global, uma vez que o número de pessoas infectadas pelo novo vírus estava tomando alta crescente, se alastrando assim em vários países e regiões do mundo. Diante deste problema que a população está vivendo, a sociedade tem procurado soluções e inovações afins de amenizar os casos e também a propagação do novo coronavírus. Desta forma, este projeto possui como finalidade a execução da fundamentação teórica e a prática efetivando assim um completo processo de ensino e aprendizagem.

Os conceitos adquiridos são utilizados para apresentar um sistema em linguagem C, afim de auxiliar hospitais e unidades de pronto atendimento (UPA) a realizarem o cadastro de pacientes que possam ser diagnosticados com COVID-19 e que necessitam de um acompanhamento para que essa informação possa ser enviada para a Secretária de Saúde, por meio do CEP e idade do indivíduo.

Por fim, este projeto visa ajudar essas entidades a catalogarem a quantidade e também a distribuição dos casos de COVID-19 que vem sendo apresentados na região.

2. A LINGUAGEM C

Dennis Ritchie é a pessoa responsável pela criação da linguagem C nos laboratórios da Bell Telephone em 1972 e criada com um propósito: ser usada no desenvolvimento de uma nova versão do sistema operacional Unix. A primeira versão do Unix utilizava Assembly. Então podemos dizer que desde o principio C foi uma linguagem criada por programadores para programadores e com uma facilidade de uso.

A linguagem C é considerada de propósito geral, ou seja é uma linguagem capaz de ser usada para praticamente qualquer tipo de projeto.

É extremamente portátil, ou seja um programa escrito em linguagem C pode ser facilmente usado em qualquer plataforma, podemos criar sistemas operacionais, aplicativos de todos os tipos, drivers e outros controladores de dispositivos, programar microcontroladores, etc. Além de toda essa flexibilidade, C é capaz de gerar programas extremamente rápidos em tempo de execução, possui uma sintaxe simples e poderosa, com instruções de alto nível.

A linguagem C influenciou de forma direta muitas linguagens como C++, Java, C# , Objective C, e muitas outras linguagens de programação tem sua sintaxe e estruturas influenciadas por C.

C é uma linguagem extremamente popular e existem muitos compiladores C disponíveis para todas as plataformas.

3. METODOLOGIA USADA NO PROJETO

Código elaborado para cadastro de pacientes diagnosticados com o vírus Covid19.

O código possui duas funções void de nomes TelaLogin e TelaMenu inicialmente criadas.

```
1 // BIBLIOTECAS IMPORTADAS
2 #include <stdio.h>
3 #include <stdlib.h>
4 #include <locale.h>
5 #include <string.h>
6
7 int opcao = 0;
8
9 void TelaLogin();
10 void TelaMenu();
11
12 int main(void)
13 {
14     // DECLARAÇÃO DE PONTUAÇÃO E ACENTUAÇÃO EM PORTUGUÊS
15     setlocale(LC_ALL, "Portuguese");
16
17     TelaLogin();
18     TelaMenu();
19
20     return 0;
21 }
22
```

São declaradas variáveis do tipo char para registrar nome, função no hospital e cadastrar um login e senha.

O usuário deverá efetuar seu cadastro e pressionar a string 5 para continuar.

```

44 // ENTRADA DAS INFORMAÇÕES DIGITADAS PELO USUÁRIO
45 printf("\n\nPrimeiro Nome: ");
46 scanf("%s", Nome);
47
48 printf("\n\nFunção no Hospital: ");
49 scanf("%s", FuncHosp);
50
51 printf("\n\nLogin: ");
52 scanf("%s", RegLogin);
53
54 printf("\n\nSenha: ");
55 scanf("%s", RegSenha);
56
57 //FINAL DO CADASTRO DAS INFORMAÇÕES
58 printf("\n\n===== USUÁRIO CADASTRADO =====");
59
60 printf("\n\nAPORTE '5' PARA CONTINUAR: ");
61 scanf("%s", Confirma);
62
63

```

O sistema irá comparar se login e senha estão corretos. Caso o usuário não digitar corretamente será imprimido na tela uma mensagem de erro.

```

23 void TelaLogin()
24 {
25
26 //DECLARAÇÃO DE VARIÁVEIS
27 char Login [30];
28 char Senha [30];
29 char RegLogin [30];
30 char RegSenha [30];
31 char Nome [30];
32 char FuncHosp [30];
33 char Confirma [1];
34
35 // REGISTRO DE UM USUÁRIO
36 printf("\n\n=====");
37 printf("\n\n===== REGISTRO DE USUÁRIO =====");
38 printf("\n\n=====");
39
40 printf("\n\n=====");
41 printf("\n\n===== PREENCHA TODOS OS CAMPOS =====");
42 printf("\n\n=====");
43

```



```

109 // SE OS DADOS FOREM IGUAIS ELE LIMPA A TELA PARA A PRÓXIMA PARTE
110 system("cls");
111
112 //MENSAGEM DE BOAS VINDAS USANDO O NOME REGISTRADO NO COMEÇO
113 printf("\n===== Seja Bem-Vindo: '%s' =====\n\n", Nome);
114
115 printf("\nAPORTE '5' PARA VER O MENU PRINCIPAL: ");
116 scanf("%d", &Confirma);
117
118 }
119

```

Ao digitar os dados corretamente, uma mensagem de boas-vindas será imprimida na tela seguida do character nome digitado pelo usuário.

```

120 void TelaMenu()
121 {
122
123     // LIMPAR A TELA PARA A PÁGINA DO MENU PRINCIPAL
124     system("cls");
125
126     //SISTEMA DO MENU PRINCIPAL
127     FILE *pont_arq;
128     int num=0;
129     int opc;
130     int idade;
131     int dnasc;
132     int datual = 2021;
133     char dia[30];
134     char mes[30];
135     char nome[30];
136     char end[30];
137     char cep[30];
138     char cpf[30];
139     char email[30];
140     char tel[30];
141     char datadiag[30];
142     char comorb[30];
143

```

Na tela de MenuPrincipal foi declarado um ponteiro do tipo FILE para salvar todas as informações do paciente que forem registradas pelo usuário em um arquivo texto.

```

144 while(num!=4)
145 {
146     system("cls");
147     printf("=====\n");
148     printf("===== MENU PRINCIPAL =====\n");
149     printf("=====\n\n");
150     printf("1. Cadastrar paciente: \n");
151     printf("2. Consultar dados do paciente: \n");
152     printf("3. Dados do sistema: \n");
153     printf("4. Sair do sistema: \n\n");
154     printf("Digite a opção: ");
155     scanf("%d", &num);
156

```

Após pressionar 5 para continuar. O usuário terá 4 opções do tipo inteiro imprimidas a tela.

```

157     switch(num) {
158
159         case 1:
160
161             // LIMPAR A TELA PARA A PÁGINA DO CADASTRO DE PACIENTES
162             system("cls");
163
164             //SISTEMA PARA CADASTRAR AS INFORMAÇÕES DO PACIENTE
165             printf("=====\\n");
166             printf("===== CADASTRAR PACIENTE =====\\n");
167             printf("=====\\n\\n");
168
169             printf("NOME COMPLETO: ");
170             fflush(stdin);
171             scanf("%[\\n]s", nome);
172
173             printf("\\nCPF: ");
174             fflush(stdin);
175             scanf("%[\\n]s", cpf);
176
177             printf("\\nDATA DE NASCIMENTO: \\n");
178             printf("DIA: ");
179             fflush(stdin);
180             scanf("%[\\n]s", dia);

```

```

205             printf("\\nDATA DO DIAGNÓSTICO: ");
206             fflush(stdin);
207             scanf("%[\\n]s", datadiag);
208
209             //MENU PARA OPÇÃO DE COMORBIDADE, SE SIM DIGITAR QUAL É, SE NÃO , FINALIZA O CADASTRO
210             while(opc!=2)
211             {
212                 printf("\\nAPRESENTA COMORBIDADE?\\n ");
213
214                 printf("> 1. SIM\\n");|
215                 printf("> 2. NÃO\\n");
216                 printf("\\nDigite a opção: ");
217                 scanf("%d", &opc);
218
219                 switch(opc){
220
221                     case 1:
222
223                         printf("Comorbidade: ");
224                         fflush(stdin);
225                         scanf("%[\\n]s", comorb);
226
227                     }
228                     break;
229
230             }

```

O sistema irá registrar as informações digitadas pelo usuário.

Após registrar todas as informações o usuário deverá informar se o paciente possui comorbidade. Selecionar se sim ou não utilizando 1 ou 2 do tipo inteiro que será armazenado no endereço opc e digitar qual a comorbidade caso possuir.

```

231
232     // IMPRESSÃO DO RELATÓRIO DO PACIENTE CADASTRADO
233     pont_arq = fopen("RELATÓRIO COVID.txt", "a");
234     fprintf(pont_arq, "PACIENTE: \\n", );
235     fprintf(pont_arq, "NOME COMPLETO: %s \\n", nome),
236     fprintf(pont_arq, "CPF: %s \\n", cpf),
237     fprintf(pont_arq, "DATA DE NASCIMENTO: %s/%s/%d \\n", dia, mes, dnasc);
238     fprintf(pont_arq, "ENDEREÇO: %s \\n", end),
239     fprintf(pont_arq, "CEP: %s \\n", cep),
240     fprintf(pont_arq, "EMAIL: %s \\n", email),
241     fprintf(pont_arq, "TELEFONE: %s \\n", tel),
242     fprintf(pont_arq, "DATA DO DIAGNÓSTICO: %s \\n", datadiag),
243     fprintf(pont_arq, "COMORBIDADE: %s \\n", comorb),
244     fclose(pont_arq);
245
246     // CÁLCULO DA IDADE DO PACIENTE
247     idade = datual - dnasc;
248
249     //FUNÇÃO DE COMPARAÇÃO DE IDADE E COMORBIDADE
250     if((idade >= 65) || (opc == 1))
251     {
252         printf("\\n\\nVOCÊ ESTÁ NO GRUPO DE RISCO!!!");
253

```

O sistema irá salvar informações do paciente em um arquivo texto, em seguida irá calcular a idade e realizar um teste para identificar se o paciente tem idade maior ou igual a 65 e possui alguma comorbidade sendo classificado dentro do grupo de risco.

Caso o paciente seja ou não do grupo de risco o sistema irá salvar seu nome, cep e idade em um arquivo texto, e imprimirá a mensagem de confirmação após o fim do cadastro.

```

254 //SE ESTIVER NO GRUPO DE RISCO SALVA UM .TXT
255 pont_arq = fopen("RELATÓRIO GRUPO DE RISCO.txt", "a");
256 fprintf(pont_arq, "PACIENTE: \n"),
257 fprintf(pont_arq, "NOME: %s\n", nome);
258 fprintf(pont_arq, "CEP: %s\n", cep),
259 fprintf(pont_arq, "IDADE: %d\n\n", idade),
260 fclose(pont_arq);
261 }
262 else
263 {
264     printf("\n\nVOCÊ NÃO ESTÁ NO GRUPO DE RISCO!!!");
265
266     //SE NÃO ESTIVER NO GRUPO DE RISCO SALVA UM .TXT
267     pont_arq = fopen("RELATÓRIO FORA DO GRUPO DE RISCO.txt", "a");
268     fprintf(pont_arq, "PACIENTE: \n"),
269     fprintf(pont_arq, "NOME: %s\n", nome);
270     fprintf(pont_arq, "CEP: %s\n", cep),
271     fprintf(pont_arq, "IDADE: %d\n\n", idade),
272     fclose(pont_arq);
273 }
274
275 printf("\n\nCADASTRO REALIZADO COM SUCESSO...\n\n");
276
277 system("pause");
278 break;
279
280
281 case 2:
282
283 // LIMPAR A TELA PARA A PÁGINA DE CONSULTA DE PACIENTES
284 system("cls");
285
286 // SISTEMA PARA CONSULTAR O CADASTRO REALIZADO
287 printf("===== \n");
288 printf("===== CONSULTA PACIENTE ===== \n");
289 printf("===== \n\n");
290
291 printf("NOME COMPLETO: %s \n", nome);
292 printf("DATA DE NASCIMENTO: %s/%s/%s \n", dia, mes, dnasc);
293 printf("CPF: %s \n", cpf);
294 printf("ENDEREÇO: %s \n", end);
295 printf("CEP: %s \n", cep);
296 printf("EMAIL: %s \n", email);
297 printf("TELEFONE: %s \n", tel);
298 printf("DATA DO DIAGNÓSTICO: %s \n", datadiag);
299 printf("COMORBIDADE: %s \n\n", comorb);
300
301 system("pause");
302 break;

```

Na tela de menu principal, selecionando a opção 2 o usuário poderá consultar os dados cadastrados do paciente.

E por fim, selecionando a opção 3, o usuário poderá consultar os dados do sistema.

```

303
304
305
306
307
308
309
310
311
312
313
314
315
316
317
318
319
320
321
322
323
324
325
326
327
328

case 3:
// LIMPAR A TELA PARA A PÁGINA DE DADOS DO SISTEMA
system("cls");
//INFORMAÇÕES SOBRE O SISTEMA
printf("===== DADOS DO SISTEMA =====\n");
printf("===== DADOS DO SISTEMA =====\n");
printf(" UM SISTEMA CRIADO PARA CADASTRAR PACIENTES COM TESTE POSITIVO DE COVID-19,\n");
printf(" ONDE O ENFERMEIRO(A) IRÁ SE CADASTRAR NO SISTEMA, LOGAR E APÓS A CONFIRMAÇÃO\n");
printf(" PODERÁ CADASTRAR E CONSULTAR OS DADOS DO PACIENTE, CASO ELE SEJA DO GRUPO DE\n");
printf(" RISCO SALVARÁ UM ARQUIVO .TXT COM O NOME, CEP E IDADE DO PACIENTE!!!\n");
printf(" NOME DOS PARTICIPANTES \n");
printf(" GABRIEL SILVA DOMINGUES - MATRÍCULA: 0446212\n");
printf(" JOÁS ALMEIDA DOS SANTOS - MATRÍCULA: 0448739\n");
printf(" SCHNEIDER SIMILIEN - MATRÍCULA: 0448458\n");
printf(" LARISSA DA COSTA CARVALHO - MATRÍCULA: 0446897\n");
printf(" APARECIDO ANGELO DA CRUZ - MATRÍCULA: 0446304\n");
printf(" VINICIUS FERNANDES SILVA - MATRÍCULA: 0448848\n");
printf(" PIM IV - ANÁLISE E DESENVOLVIMENTO DE SISTEMAS - UNIP \n");

system("pause");
break;
}
}

```

4. SOBRE O PROJETO (VARIÁVEIS E COMANDOS)

Para a realização do projeto em questão fizemos uso de algumas variáveis e comandos onde serão mencionadas abaixo. Como também sua finalidade dentro do código:

stdio.h: É um cabeçalho da biblioteca padrão do C. Seu nome vem da expressão inglesa standard input-output header, que significa "cabeçalho padrão de entrada/saída".

stdlib.h: É um arquivo cabeçalho da biblioteca de propósito geral padrão da linguagem de programação C. Ela possui funções envolvendo alocação de memória, controle de processos, conversões e outras

string.h: É um arquivo cabeçalho que fornece funções, macros e definições da biblioteca padrão da linguagem de programação C para manipulação de cadeias de caracteres e regiões de memória.

void: Ele nos permite fazer funções que não retornam nada e funções que não têm parâmetros.

char: É usado para armazenar o valor inteiro de um membro do conjunto de caracteres.

printf: Esta função formata e imprime uma série de caracteres e valores para o fluxo de saída padrão (tela do computador)

scanf: É uma família de funções da linguagem de programação C disponibilizada pelo arquivo cabeçalho `stdio.h` que permite a leitura de dados a partir de uma fonte de caracteres de acordo com um formato pré determinado

while: A função do laço `while` é repetir um determinado trecho ENQUANTO uma determinada condição for verdadeira.

strcmp: Esta função compara duas strings e devolve um valor inteiro que lhe diz qual das strings vem antes no código ASCII: Sintaxe: `strcmp (string1, string2);`

switch: Compara a mesma variável ou expressão com várias opções

fflush(stdin): É usado para limpar o buffer do teclado

fprintf: É usado para imprimir conteúdo em arquivo em vez do console.

fopen: Esta função abre um arquivo em disco para gravar ou ler.

fclose: Fecha um arquivo aberto pela função `fopen`

if / else: É uma estrutura de decisão examina uma ou mais condições e decide quais instruções serão executadas dependendo se a condição for verdadeira ou falsa.

5. MANUAL DE INSTALAÇÃO

O executável do arquivo está em uma pasta em formato zip, para extrair-lo será necessário um programa de extração de arquivos, como 7-zip ou Winrar.

Ao extrair-lo para uma pasta, veremos três arquivos, o primeiro sendo um arquivo em formato `.c`, o segundo sendo em formato `.exe` e o terceiro em formato `.obj`.

Para que o programa seja executado e suas funcionalidades validadas devemos abrir o `.exe`

6. SISTEMA DE CADASTRO DE PACIENTES COM COVID-19

É um software desenvolvido em C que visa coletar, manter e recuperar informações pertencentes à categoria de Sistema (SPT).

Nosso SOFTWARE está dividido em 2 partes, uma para idosos com COVID-19 e pacientes que apresentam alguma comorbidade, como: diabetes, hipertensão e

outros também com COVID-19 (GRUPO DE RISCO) e a outra parte para pacientes jovens e adultos com COVID-19.

Em nosso software coletamos as informações de cada paciente como endereço completo, nome completo, data de nascimento, documentos (CPF), telefone ou e-mail de contato e data de diagnóstico

E depois um questionário para ver se esse paciente faz parte do “grupo de risco” de pessoas que têm doenças como diabetes, tuberculose e hipertensão. Após a coleta de dados, automaticamente cada paciente é classificado na parte do software designada.

SPT tem por definição sistemas tradicionais que coletam, guardam, modificam e recuperam as transações, gerenciando as informações das organizações. Uma transação é um evento que gera ou modifica dados.

E também o SPT tem por características tempo de resposta rápida – atender os usuários, confiabilidade, flexibilidade, processamento controlado – suporte à organização, armazenamento e recuperação de informação.

6.1 REGISTRO DO USUÁRIO

O usuário ao executar o programa deverá preencher os campos apresentados em tela, como: PRIMEIRO NOME, FUNÇÃO NA ENTIDADE DE TRABALHO como também um LOGIN e SENHA para acesso ao sistema.

Sendo cada opção preenchida devendo ter a tecla ENTER como confirmação para seguir ao próximo campo.

Após preenchido as opções o sistema retorna a mensagem de USUARIO CADASTRADO e assim o mesmo pode prosseguir para o acesso na área de LOGIN.

FIGURA 1: TELA DE REGISTRO DO USUÁRIO



FONTE: PRINTSCREEN DO PROGRAMA EM SISTEMA OPERACIONAL WINDOWS

6.2 TELA DE LOGIN

O usuário após cadastrar suas credenciais na tela de registro o sistema lhe dá acesso a área de login conforme imagem abaixo.

Nesta etapa o usuário preenche as credenciais cadastradas com seu LOGIN e SENHA e em seguida confirma os dados, assim tendo acesso ao menu principal do programa.

FIGURA 2: TELA DE LOGIN



FONTE: PRINTSCREEN DO PROGRAMA EM SISTEMA OPERACIONAL WINDOWS

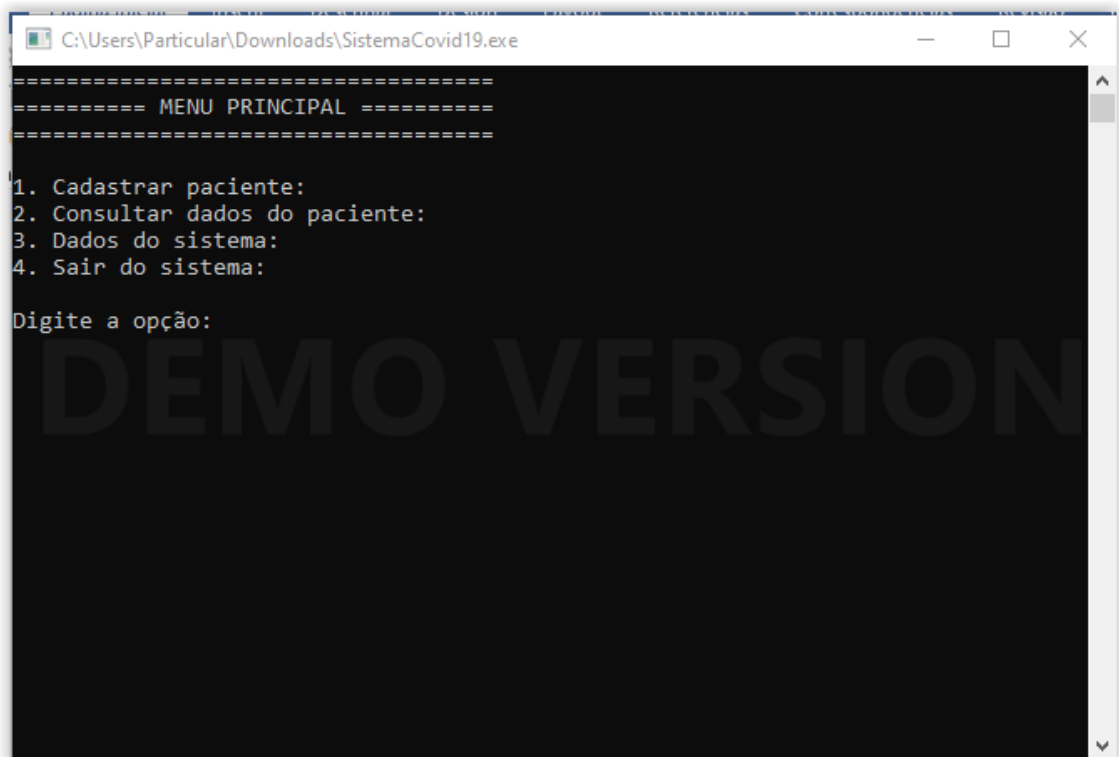
6.3 MENU PRINCIPAL

Nesta etapa o usuário já está autorizado a utilizar o sistema de cadastro e tem a sua disposição as seguintes opções:

1. **CADASTRAR PACIENTE:** área para cadastro completo do paciente em questão.
2. **CONSULTAR DADOS DO PACIENTE:** área de consulta rápida do ultimo cadastro realizado no sistema.
3. **DADOS DO SISTEMA:** área contendo informações sobre o sistema de cadastro, como também os integrantes responsáveis por sua elaboração e desenvolvimento.
4. **SAIR DO SISTEMA:** opção em que o usuário encerra o sistema, logo o sistema é fechado.

Para acessar as opções o usuário deve apenas informar ao sistema a opção desejada e digitar o numero pertinente a cada opção, seguido da tecla ENTER.

FIGURA 3: TELA DO MENU PRINCIPAL



FONTE: PRINTSCREEN DO PROGRAMA EM SISTEMA OPERACIONAL WINDOWS

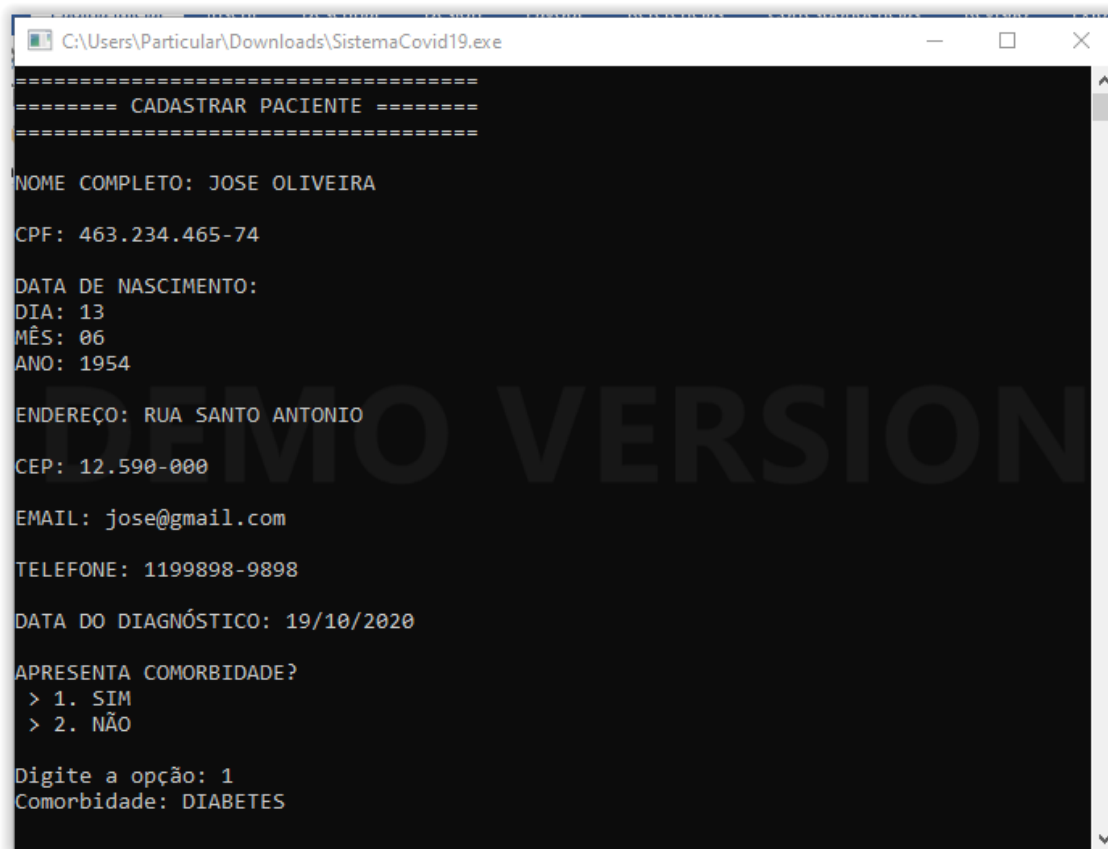
6.4 CADASTRO DO PACIENTE

O usuário acessando a primeira opção do sistema terá acesso a área de cadastro de paciente. Nesta tela (figura 4) o usuário deverá informar ao sistema os dados pessoais do paciente, como também data do diagnóstico e caso tenha alguma comorbidade.

Uma vez o usuário informa esses dados ao sistema o mesmo realiza a verificação e validação do paciente, caso ele seja do grupo de risco, tendo como base o calculo de sua idade como também se há algum tipo de comorbidade. Tendo essas duas informações confirmadas o sistema gera dois arquivos .txt (figura 4.1) em ambiente externo (diretório do sistema operacional).

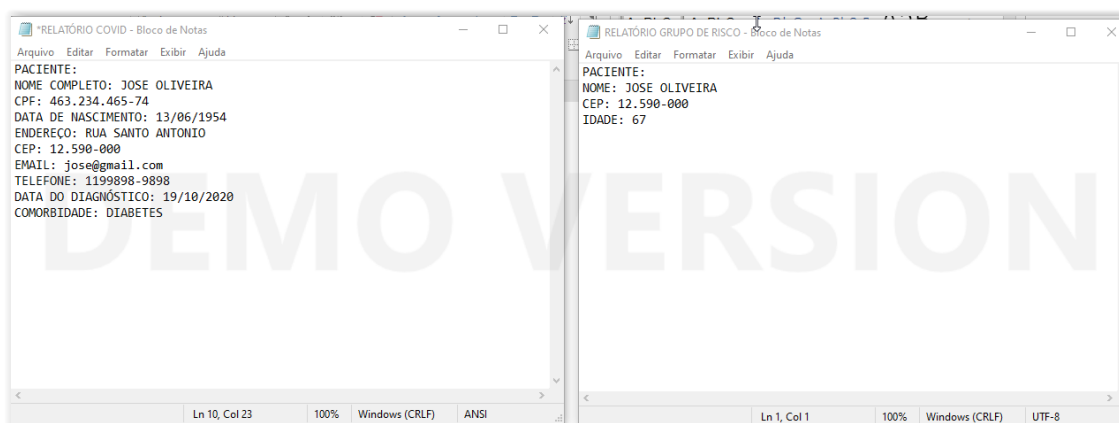
Um arquivo com todas as informações do paciente e também um arquivo caso ele seja do grupo de risco, contendo CEP e IDADE. Ambos os arquivos são enviados para a Secretaria da Saúde da cidade.

FIGURA 4: TELA DE CADASTRO DE PACIENTE



FONTE: PRINTSCREEN DO PROGRAMA EM SISTEMA OPERACIONAL WINDOWS

FIGURA 4:1 ARQUIVOS .txt GERADO PELO SISTEMA DE CADASTRO

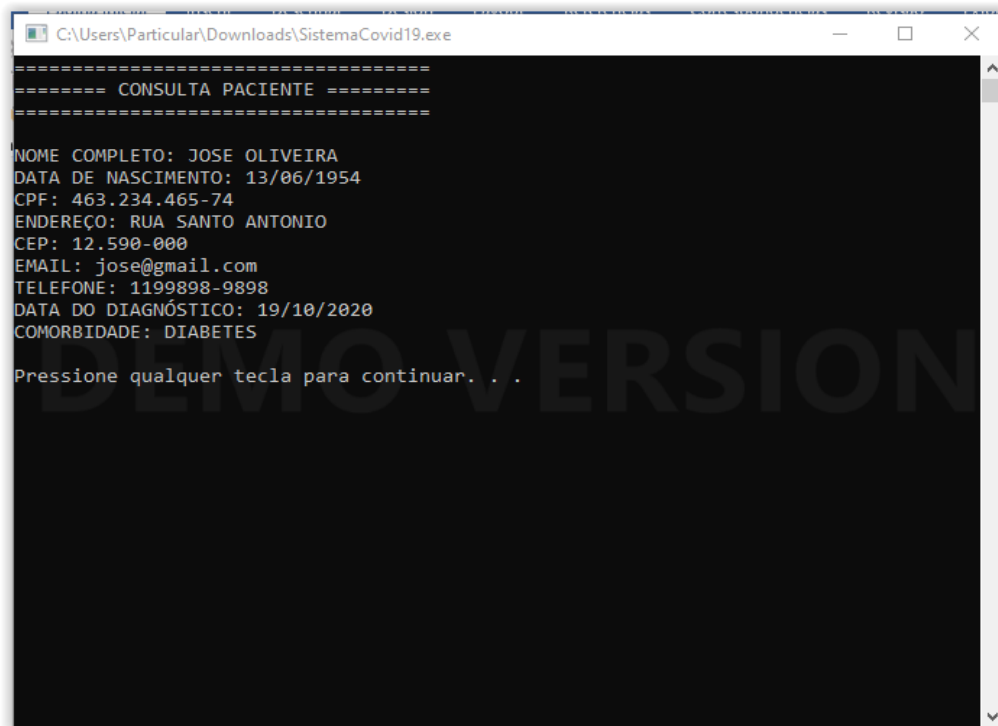


FONTE: PRINTSCREEN DO BLOCO DE NOTAS EM SISTEMA OPERACIONAL WINDOWS

6.5 CONSULTA DE PACIENTE

Ao acessar a opção 2 do sistema, o usuário tem acesso ao cadastro realizado recentemente, assim o mesmo pode realizar uma consulta rápida dos dados fornecidos ao sistema.

FIGURA 5: TELA DE CONSULTA DO PACIENTE

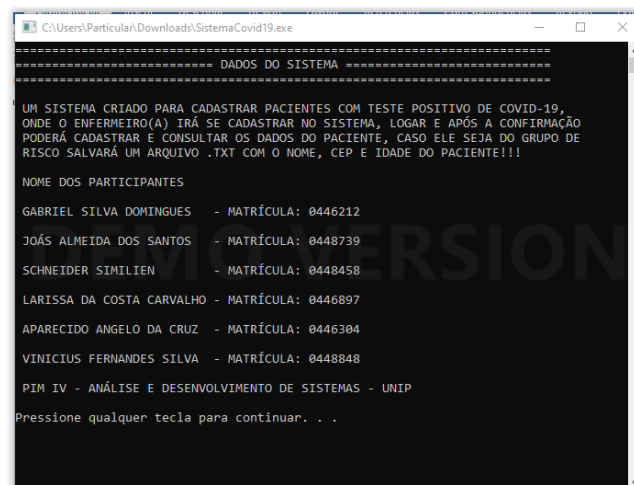


FONTE: PRINTSCREEN DO PROGRAMA EM SISTEMA OPERACIONAL WINDOWS

6.6 DADOS DO SISTEMA

Nesta área do usuário pode consultar uma breve descrição sobre a finalidade do sistema, como também as pessoas responsáveis pelo seu desenvolvimento.

FIGURA 6: TELA DE DADOS DO SISTEMA



FONTE: PRINTSCREEN DO PROGRAMA EM SISTEMA OPERACIONAL WINDOWS

6.7 SAIR

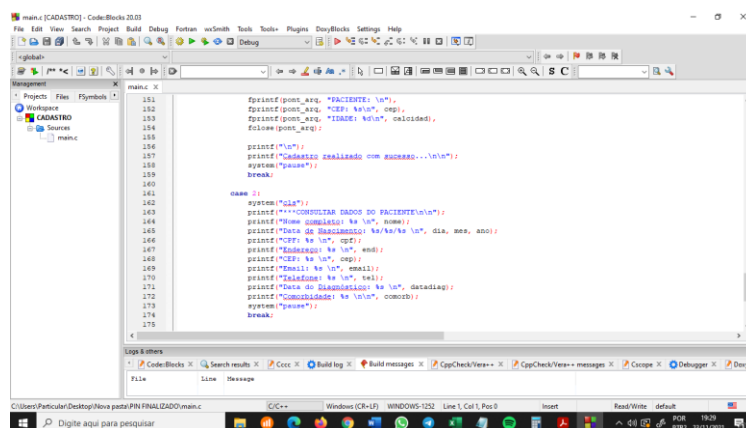
Para sair do sistema, basta digitar o número 4, sendo assim logo que o usuário digitar a opção o sistema fecha em seguida.

7. COMPILAÇÃO DO PROGRAMA

O projeto desenvolvido foi inteiramente codificado no CODE::BLOCKS e neste mesmo programa realizamos a compilação do mesmo, como também todos os testes para a conclusão do projeto final.

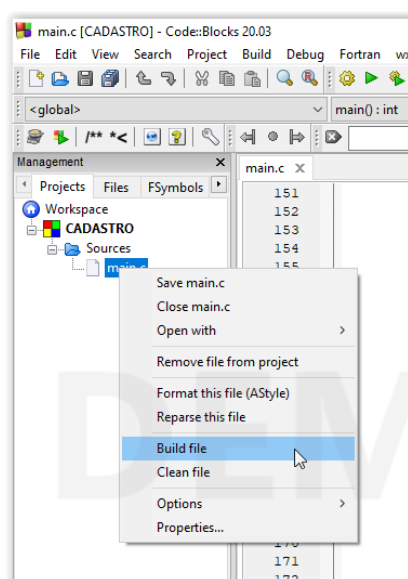
Abaixo a forma como o código pode ser compilado no programa.

FIGURA 7: TELA DA ESTRUTURA DO CÓDIGO NO CODE BLOCKS



**FONTE: PRINTSCREEN DO PROGRAMA CODE:BLOCKS EM SISTEMA OPERACIONAL
WINDOWS**

FIGURA 8: TELA DA ESTRUTURA DO CÓDIGO NO CODE BLOCKS (COMPILAR)



FONTE: PRINTSCREEN DO PROGRAMA
CODE:BLOCKS EM SISTEMA OPERACIONAL WINDOWS

8. CONCLUSÃO

Com base nas disciplinas Linguagem e Técnicas de Programação e também Engenharia de Software I estudadas ao longo do semestre, pudemos estruturar o projeto num todo, de forma com que passamos a compreender desde a ideia inicial do projeto até sua conclusão, passando por todas as etapas necessárias para a realização do código.

O desafio proposto em estruturar um código para que as entidades hospitalares e centros de diagnóstico do COVID-19 pudessem utilizar foi de grande valia, visto no cenário em que o mundo se encontra com o impacto causado pelo vírus.

O sistema criado visa a agilidade no reporte de pessoas que precisam ser diagnosticadas com certa agilidade. Com isso o projeto tem esse suporte necessário.

9. REFERÊNCIAS

- CPROGRAMMING. CPROGRAMMING, 2021. Página Inicial. Disponível em: <<https://cboard.cprogramming.com/>>. Acesso em: 10/11/2021.
- AULAS INTERATIVAS. UNIP, 2021. PLATAFORMA AVA. Disponível em: <https://ava.ead.unip.br/webapps/portal/execute/tabs/tabAction?tab_group_id=101>. Acesso em: 04/10/2021 - 22/11/2021
- warning: passing argument 1 of ‘memcpy’ makes pointer from integer without a cast. CPROGRAMMING, 2009. Getting started with C or C++. Disponível em: <<https://cboard.cprogramming.com/c-programming/119016-warning-passing-argument-1-%91memcpy%92-makes-pointer-integer-without-cast.html>>. Acesso em: 12/11/2021
- Comando de Controle Switch Case. EMBARCADOS, 20/10/2016. Comando de Controle Switch Case. Disponível em: <<https://www.embarcados.com.br/comando-de-controle-switch-case/>>. Acesso em: 12/11/2021
- Menu com Switch em C. CLUBE DO HARDWARE, 02/06/2019. Menu com Switch em C. Disponível em: <<https://www.clubedohardware.com.br/topic/1376122-menu-com-switch-em-c/>>. Acesso em: 13/11/2021
- Linguagem C. INTELLECTUALE TECNOLOGIA & TREINAMENTO, 02/06/2019. Estrutura de decisão if em Linguagem C. Disponível em: <<http://linguagemc.com.br/estrutura-de-decisao-if-em-linguagem-c/>>. Acesso em: 14/11/2021
- O teste condicional SWITCH: o que é, para que serve e como usar o switch em C. C PROGRESSIVO. SWITCH em C: o que é e como usar o comando. Disponível em: <<https://www.cprogressivo.net/2013/02/O-teste-condicional-SWITCH-o-que-e-para-que-serve-e-como-usar-o-switch-em-C.html>>. Acesso em: 14/11/2021

- O teste condicional SWITCH: o que é, para que serve e como usar o switch em C. C PROGRESSIVO. O Switch sem o break. Disponível em: <https://www.cprogressivo.net/2013/02/O-teste-condicional-SWITCH-o-que-e-para-que-serve-e-como-usar-o-switch-em-C.html>>. Acesso em: 17/11/2021
- Fazendo testes e comparações - operador de igualdade (==), maior (>), menor (<), maior igual (>=), menor igual (<=), de diferença (!=) e de módulo, ou resto da divisão (%). C PROGRESSIVO. Testar e comparar, testar e comparar...a essência da programação. Disponível em: <https://www.cprogressivo.net/2013/02/Fazendotestesecomparacoes-operadordeigualdademaioresmenorigualmenorigualedeiferencaedemoduloourestoda-divisao.html>>. Acesso em: 18/11/2021