

UNIVERSIDADE PAULISTA – UNIP EaD

Projeto Integrado Multidisciplinar

Curso Superior de Tecnologia em Análise e Desenvolvimento de Sistemas

Ewerton Siqueira Costa Brandão (RA 2032437)
Marcio Costa De Oliveira (RA 2050795)
Danillo Raphael Rodrigues Macor (RA 2077159)
Davi Edimar Paciência da Silva (RA 2068210)
Weverton Luis Franzoso (RA 2037835)
Rodrigo Venâncio Da Silva (RA 2004486)

Sistema em C para cadastrar pacientes diagnosticados com covid-19

LUZIÂNIA-GO
2020

UNIVERSIDADE PAULISTA – UNIP EaD

Projeto Integrado Multidisciplinar

Curso Superior de Tecnologia em Análise e Desenvolvimento de Sistemas

Sistema em C para cadastrar pacientes diagnosticados com covid-19

Projeto multidisciplinar para o desenvolvimento e aplicações dos conhecimentos adquiridos nas disciplinas cursadas no segundo bimestre vigente no 2º semestre do curso de Análise e desenvolvimento de sistemas na instituição de ensino Universidade paulista - UNIP EaD.

LUZIÂNIA-GO
2020

SUMÁRIO

1 INTRODUÇÃO	6
2 ENGENHARIA DE SOFTWARE	7
2.1 Metodologia	7
2.1.1 Metodologia cascata	7
2.2 Requisitos	9
2.2.1 Requisito funcional	10
2.1.2 Requisito não funcional.....	11
3 DESENVOLVIMENTO	11
3.1 Linguagem C	12
3.2 Funcionamento do sistema	13
3.3 Arquivos.....	18
3.4 Execução do sistema.....	19
3.5 Fluxograma	21
4 CONCLUSÃO	22
REFERÊNCIAS	23

RESUMO

Este trabalho tem como o principal objetivo, desenvolver um sistema em linguagem C para ser utilizado nos hospitais em meio a pandemia para cadastrar pacientes que foram diagnosticados com o covid-19 e necessitam de acompanhamento médico durante o período de internação.

Devido a pandemia global de covid-19, em paralelo surgiu a necessidade e demanda de inovação tecnológica para ambientes hospitalares ao redor do mundo, é importante alinhar o uso da tecnologia com a situação global para identificação e rastreamento das pessoas contaminadas com o vírus, muitos governos ao redor do mundo estão usando a tecnologia diante da pandemia para capacitação de informações sobre a propagação do vírus e auxiliar os profissionais de saúde na luta contra esta temível doença que vem assolando o mundo.

Para resolução do caso e para o desenvolvimento do sistema serão utilizadas as disciplinas vigentes do bimestre, Linguagem e Técnicas de Programação e Engenharia de Software I.

Palavras-chaves: Linguagem C, Sistema, Covid-19, Monitoramento, Profissionais de Saúde, Pandemia.

ABSTRACT

This study has as its main objective, to develop a C-language system to be used in hospitals in the middle of the pandemic to register patients who were diagnosed with covid-19 and require medical follow-up during the hospitalization period.

Due to the global pandemic of covid-19, in parallel arose the need and demand for technological innovation for hospital environments around the world, it is important to align the use of technology with the global situation to identify and track people infected with the virus, many governments around the world are using the technology in the face of the pandemic to enable information on the spread of the virus and assist health professionals in the fight against this fearsome disease that comes besoling the world.

For case resolution and system development will be used the current disciplines of the bimester, Language and Programming Techniques and Software Engineering I.

Keywords: Language C, System, Covid-19, Monitoring, Health Professionals, Pandemic.

1 INTRODUÇÃO

Devido a pandemia global de covid-19 e o aumento de casos de contaminação, ocorreu a necessidade da criação de um sistema operacional que tem como função cadastrar pacientes diagnosticados com a covid-19 que necessitam de acompanhamento médico, o sistema ainda terá a função de enviar dados e informações desses pacientes para a Secretária de Saúde, esta é a problemática do projeto.

Como pouco se sabe sobre a covid-19 muitos órgãos de saúde ao redor do mundo estão utilizando a tecnologia para obter dados na luta e prevenção do vírus, devido a grande proporção de infecção do vírus. Os primeiros casos diagnosticados foram na China, mais especificamente na província de Wuhan ainda em 2019 no dia 1º de dezembro. Fazendo uma alusão ao ano de 2019, o sistema desenvolvido neste projeto foi apelidado de CAD 19 (CAD de cadastro e 19 do termo Covid-19, em referência ao ano de 2019 onde os primeiros casos apareceram na província de Wuhan na China).

No decorrer do desenvolvimento deste projeto, vamos apresentar as soluções para resolução da problemática, utilizando os conceitos da disciplina Linguagem e Técnicas de Programação, para concepção do sistema fazendo uso da Linguagem C, vamos apresentar as principais características e funções utilizadas para o desenvolvimento e com o auxílio da disciplina Engenharia de Software I, onde foi possível ter uma base para implementação de uma metodologia na criação deste sistema.

Este projeto tem o objetivo de desenvolver um sistema de cadastro, captação de dados e compartilhamento de informações para a central da secretária da saúde durante acompanhamento médico de pacientes contaminados com covid-19 e auxiliando os profissionais de saúde.

2 ENGENHARIA DE SOFTWARE

Engenharia de software é uma área da computação voltada à especificação, desenvolvimento, manutenção e criação de software, com a aplicação de tecnologias e práticas de gerência de projetos e outras disciplinas, visando organização, produtividade e qualidade. Atualmente, essas tecnologias e práticas englobam linguagens de programação, banco de dados, ferramentas, plataformas, bibliotecas, padrões de projeto de software, processo de software e qualidade de software. Além disso, a engenharia de software deve oferecer mecanismos para se planejar e gerenciar o processo de desenvolvimento de um sistema computacional de qualidade e que atenda às necessidades de um requisitante de software.

2.1 Metodologia

As metodologias de desenvolvimento de software consistem, basicamente, no conjunto de abordagens que podem ser utilizadas para a criação de sistemas de processamento de dados.

O sucesso de qualquer projeto voltado à elaboração de software depende diretamente da escolha da metodologia mais adequada.

Para garantir mais eficiência em todo o desenvolvimento, cabe ao responsável pela equipe conhecer as diferentes metodologias disponíveis e optar por aquela que seja melhor para o seu caso.

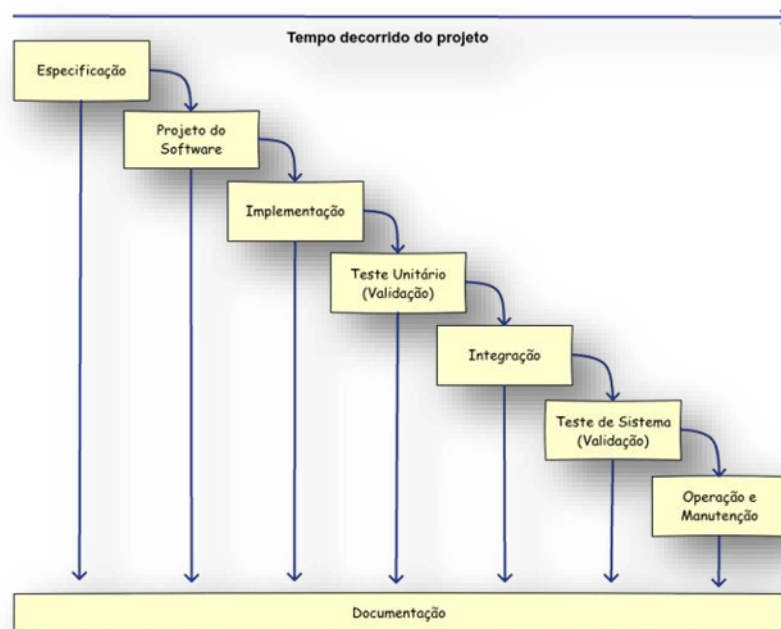
É evidente que cada método possui vantagens e desvantagens, por isso é indispensável conhecer as metas, os objetivos, prazos e orçamentos envolvidos no projeto para determinar qual é a abordagem mais alinhada.

2.1.1 Metodologia cascata

Nesse projeto adotamos a Metodologia Cascata. Criado em 1970 por Winston Walker Royce, o Modelo em Cascata (também conhecido como Modelo Sequencial Linear) é o mais antigo de todos os processos e possui esse nome devido a sua forma sequencial cascadeada que acontece de uma fase para a outra.

Considerado o “modelo clássico de desenvolvimento”, o Modelo em Cascata é “sugerido para projetos pequenos” e sua ideia defende que “para uma fase iniciar, a anterior deve estar totalmente finalizada” (ENGHOLM, 2010) e que o “resultado de cada etapa é a aprovação de um ou mais documentos ‘assinados’” (SOMMERVILLE, 2011).

IMAGEM 01 – O modelo em Cascata



Fonte: <https://medium.com/contexto-delimitado/o-modelo-em-cascata-f2418addaf36>

Basicamente, o desenvolvimento é dividido em cinco etapas, que começam assim que a anterior termina:

1. Especificação: é feita a análise e captação das necessidades do cliente e definidos os “requisitos do sistema” com suas funcionalidades;
2. Projeto do software: é elaborado o desenho da arquitetura geral do sistema, seguindo as informações coletadas na etapa anterior. Tudo é separado em partes chamadas de “unidades de programa” para poder agilizar o desenvolvimento. Cada unidade terá seu próprio “requisito de unidade”;
3. Implementação: nesta etapa, com o projeto definido, cada “unidade de programa” é implementada, podendo alocar vários programadores simultaneamente.

4. Teste unitário: no final da implementação, são feitos os “testes de unidade” para certificar (validação) que as implementações atendem aos “requisitos de unidade”;
5. Integração: com as unidades testadas, a próxima tarefa é integrar todas elas para compor um sistema completo.
6. Teste de Sistema: no final da integração, é feito um teste geral para certificar (validação) que os “requisitos do sistema” foram completamente atendidos. Por fim, o software é entregue ao cliente;
7. Operação e manutenção: Ao operar, ou seja, utilizar o sistema, o cliente poderá encontrar erros ou necessidades que não foram identificadas anteriormente. Esta fase se responsabiliza pela correção dos erros ou melhorias do sistema (evolução).

Como o nosso grupo é composto por 6 integrantes , após a análise de requisitos “quebramos” o projeto em 3 partes onde cada um par de integrantes ficou responsável por uma parte.

- PARTE I Um par de programadores ficou responsável por implementar o cadastro de usuário, senha, login no sistema e as devidas validações
- PARTE II O segundo par de programadores ficou responsável por implementar o cadastro de pacientes e as validações.
- PARTE III O terceiro par de programadores ficou responsável por implementar os menus e arquivos.txt.

2.2 Requisitos

A palavra requisito segundo o Dicionário Aurélio tem como significados condições necessárias, geralmente obrigatórias, para se conseguir algo se alcançar um propósito.

Tendo como base esse significado, os requisitos em um software vai estabelecer as condições necessárias e obrigatórias para que o resultado do software

atendam as características necessárias do pedido. E deve ser feita toda uma análise se é possível verificar se o requisito está sendo atendido ou respeitado .

Os requisitos são divididos em dois tipos, os funcionais e não funcionais:

2.2.1 Requisito funcional

O termo Requisito Funcional se refere a requisição de uma função que um software deverá atender/realizar. Ou seja, exigência, solicitação, desejo, necessidade, que um software deverá materializar.

Na engenharia de software, mais precisamente na engenharia de requisitos (comumente chamada apenas de Análise de Requisitos ou Levantamento de Requisitos) é a disciplina que identifica a “dor” do cliente, faz um “diagnóstico” sobre sua origem e propõem um “tratamento terapêutico” para curá-lo.

TABELA 01 – Requisito funcional

REQUISITO FUNCIONAL (RF)	NOME	DESCRIÇÃO
RF 01	CADASTRO DE USUÁRIO E SENHA	Área destinada a criação de um usuário e senha para a utilização do sistema
RF 02	EFETUAR LOGIN	Autenticação de usuários cadastrados no sistema , permitindo a realização de operações na área restrita do sistema
RF 03	CADASTRO DE PACIENTES	Área destinada a realização de cadastro das informações pessoais do paciente
RF 04	VALIDAR CAMPO CPF	Validar o campo de cadastro de CPF para que não haja pacientes com CPF's duplicados no sistema
RF 05	INFORMAÇÕES SALVAS EM ARQUIVO	Área externa do software onde fica salva todas as informações referentes ao paciente.

Fonte: Autoria própria

2.1.2 Requisito não funcional

Requisitos não funcionais são os requisitos relacionados ao uso da aplicação em termos de desempenho, usabilidade, confiabilidade, segurança, disponibilidade, manutenção e tecnologias envolvidas.

“Um requisito funcional é “O QUE” o sistema deve fazer. Um requisito não funcional é “COMO” o sistema deve fazer.”

TABELA 02 – Requisito não funcional

REQUISITO NÃO FUNCIONAL (RNF)	NOME	DESCRIÇÃO
RNF 01	SISTEMA OPERACIONAL	O sistema deverá rodar em qualquer plataforma.
RNF 02	LINGUAGEM DE PROGRAMAÇÃO	O desenvolvimento deve ser em linguagem C

Fonte: Autoria própria

Nesse projeto abordamos superficialmente os requisitos especificando somente o identificador, nome do requisito e descrição. É válido ressaltar que as especificações de um software ocorre de forma detalhada, como o nome do autor, data de criação, versão categorias, por exemplo.

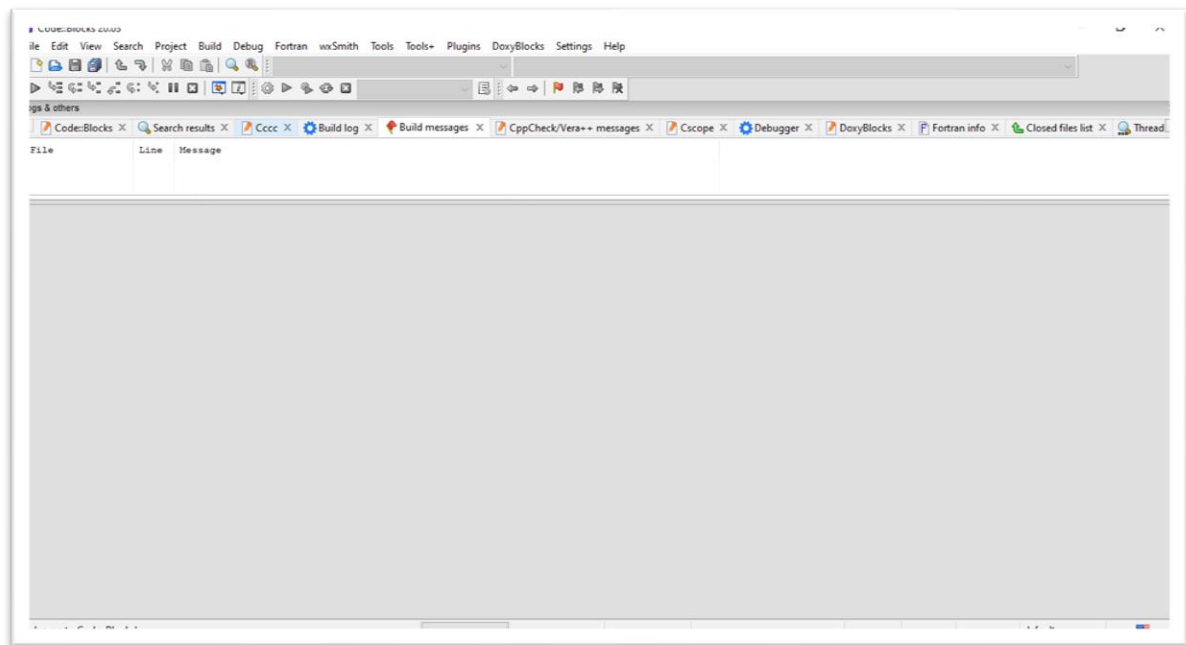
3 DESENVOLVIMENTO

Em todo processo de codificação e execução do sistema será utilizado a IDE CODE BLOCKS, utilizando a linguagem de codificação C.

Code Blocks (ou C:B) é um ambiente de desenvolvimento integrado de código aberto e multiplataforma. Ele foi desenvolvido em C++, usando wxWidgets. Sua arquitetura é orientada a plugin, de forma que suas funcionalidades são definidas pelos plugins fornecidos a ele.

Code Blocks é voltado para o desenvolvimento em C/C++ e Fortran, podendo também ser usado para a criação de ARM, AVR, D (linguagem de programação), DirectX, FLTK, GLFW, GLUT, GTK+, Irrlicht, Lightfearther, MATLAB, OGRE, OpenGL, Qt, SDL, SFML, STL, SmartWin e programas ou aplicativos com wx,.

IMAGEM 02 – Interface inicial Code Blocks



Fonte: Autoria própria

3.1 Linguagem C

A linguagem de programação C é uma linguagem de programação compilada de propósito geral, criada em 1972 por Dennis Ritchie na empresa AT&T Bell Labs para desenvolvimento do sistema operacional UNIX.

C é uma linguagem imperativa e procedural para implementação de sistemas, desenvolvida para ser uma linguagem de alto nível, para maior reaproveitamento do código.

Na linguagem C, precisamos gerenciar explicitamente a memória alocada, podemos manipular diretamente o endereço da memória, precisamos entender o conceito de passagem de parâmetros por valor e referência, o conceito de variáveis de ponteiro e assim por diante. O conhecimento que adquirimos nos permite escrever programas corretos e eficazes. Esse conhecimento é especialmente importante quando precisamos projetar novas estruturas de dados para atender a necessidades específicas. Linguagens de programação mais abstratas suportam nativamente as estruturas de dados mais comuns; na linguagem C, precisamos implementá-las. O conhecimento obtido na implementação de estruturas nos dá uma melhor compreensão de qual estrutura deve ser usada em uma determinada aplicação, nos

permite nos adaptar às estruturas existentes quando necessário e nos ensina a avaliar a eficiência dos algoritmos que utilizam essas estruturas.

3.2 Funcionamento do sistema

O sistema conta com diferentes telas, onde aparece informações e opções que pode ser acionadas clicando na tecla correspondente ao número da opção. E para avançar para uma próxima linha é necessário pressionar a tecla ENTER.

- A primeira tela do sistema é o ponto de partida para o funcionário ter acesso a área de cadastros, cadastrar usuário e senha ou sair do sistema.

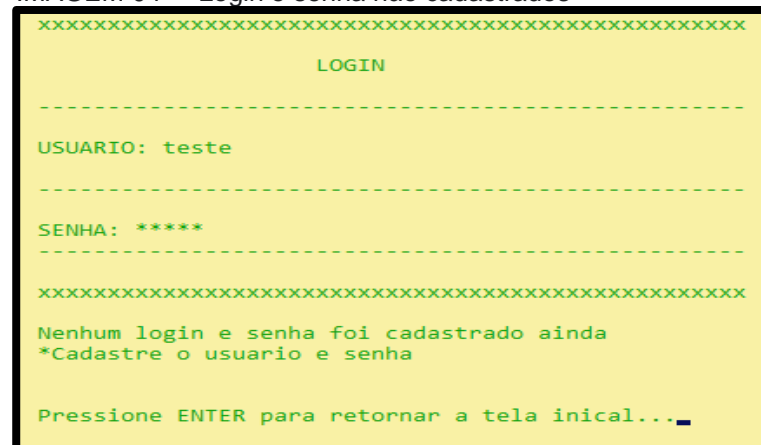
IMAGEM 03 – Tela inicial



Fonte: Autoria própria

- No primeiro uso do sistema o funcionário deve acessar a opção 2 e cadastrar o usuário e senha, caso o contrário seu acesso não é permitido.

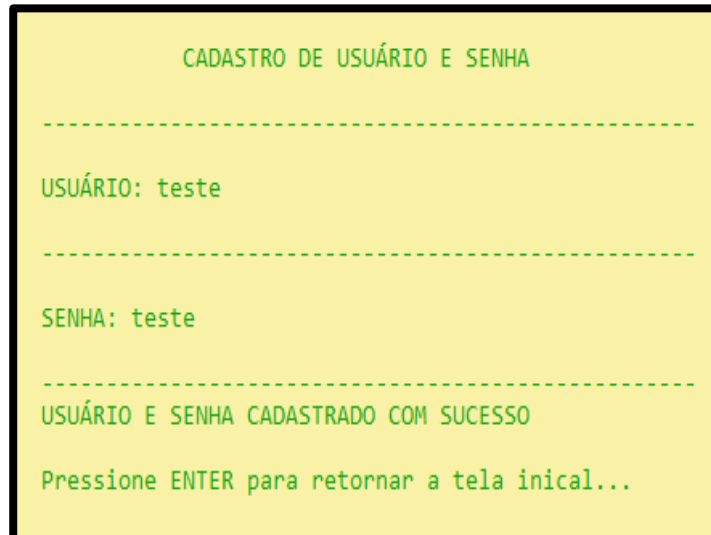
IMAGEM 04 – Login e senha não cadastrados



Fonte: Autoria própria

- Após o cadastro de usuário e senha o funcionário deve retornar ao menu inicial para efetuar o login no sistema.

IMAGEM 05 – Tela de cadastro de usuário e senha



CADASTRO DE USUÁRIO E SENHA

USUÁRIO: teste

SENHA: teste

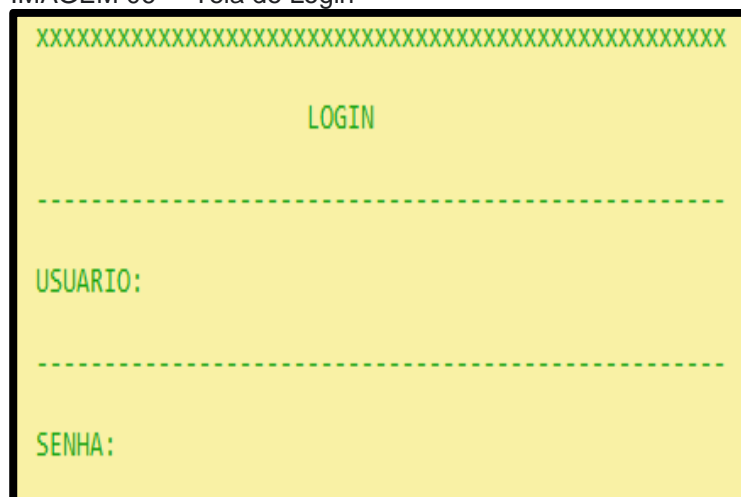
USUÁRIO E SENHA CADASTRADO COM SUCESSO

Pressione ENTER para retornar a tela inicial...

Fonte: Autoria própria

- Escolhendo a opção 1 da tela inicial o funcionário deve entrar com usuário e senha.

IMAGEM 06 – Tela de Login



XX

LOGIN

USUARIO:

SENHA:

FONTE: Autoria própria

- Caso as credenciais estejam corretas é aberta a tela com a opção de cadastrar os dados do paciente, caso o contrário, é exibido uma mensagem de erro na tela com um leve beep indicando que as credenciais estão incorretas.

IMAGEM 07 – Tela com opção de cadastrar paciente

```

XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
                                LOGIN
                                -----
USUARIO: admin
                                -----
SENHA: *****
                                -----
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX

    Logado com sucesso
                                -----
-[1]CADASTRAR PACIENTES                -
-[2]Voltar ao menu inicial              -
                                -----
    Digite o número da opção:
  
```

FONTE: Autoria própria

IMAGEM 08 – Login inválido

```

    (Usuário e/ou senha incorreto)

    (Tente novamente)

    Pressione ENTER para retornar a tela inicial...
  
```

FONTE: Autoria própria

- Escolhendo a opção 1 da tela com a opção de cadastrar paciente, o funcionário tem acesso as telas de preenchimento de dados. O primeiro campo que dever ser preenchido é o CPF. O campo CPF é responsável por validar e permitir apenas 1 cadastro de paciente por uma única numeração de CPF. Caso o sistema verifique que o CPF digitado não existe, o sistema continua solicitando os demais dados.

IMAGEM 09 – Cadastro de CPF

```

*****DADOS PESSOAIS*****
                                -----
    Digite os 11 números do cpf sem ponto e traço

    CPF: 00000000000
  
```

FONTE: Autoria própria

- Caso o funcionário tente cadastrar um novo paciente com um mesmo número de CPF já existente no sistema, é exibido uma mensagem de erro.

IMAGEM 10 – Erro ao cadastrar CPF

```
*****DADOS PESSOAIS*****
-----
Digite os 11 números do cpf sem ponto e traço
CPF: 0000000000

***Já consta no sistema um cadastro com esse número de cpf***
Digite os 11 números do cpf sem ponto e traço
CPF:
```

FONTE: Autoria própria

- Após o preenchimento do CPF o funcionário deve cadastrar os demais dados (nome, data de nascimento, e-mail, telefone, endereço ,por exemplo).

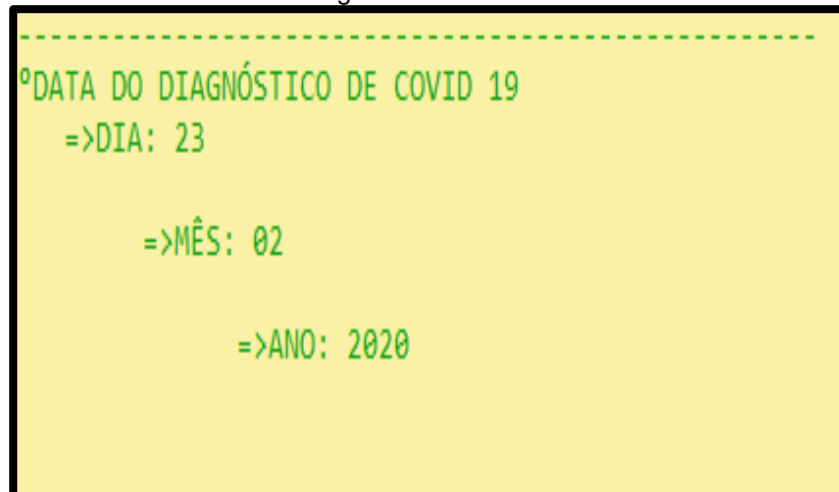
IMAGEM 11 – Preenchimento de dados pessoais

```
CPF: 010101010
°NOME: FULANO CICLANO
-----
°DATA DE NASCIMENTO
=>DIA dd : 23
=>MÊS mm : 02
=>ANO aaaa : 1950
-----
EMAIL: email@email.com
-----
°NUMERO DE TELEFONE
***Cadastre até dois números de telefone
Digite os números todos juntos XXXX...
INFORME O N° DE TELEFONE PRINCIPAL: 40028922
-----
INFORME O N° DE TELEFONE 2: 36234546_
```

FONTE: Autoria própria

- Antes prosseguir para o preenchimento dos dados de endereço é necessário informar a data de diagnóstico do covid 19.

IMAGEM 12 – Data de diagnóstico

A yellow rectangular form with a black border. At the top, there is a dashed green line. Below it, the text "ºDATA DO DIAGNÓSTICO DE COVID 19" is written in green. Underneath, the fields are filled with green text: "=>DIA: 23", "=>MÊS: 02", and "=>ANO: 2020".

ºDATA DO DIAGNÓSTICO DE COVID 19

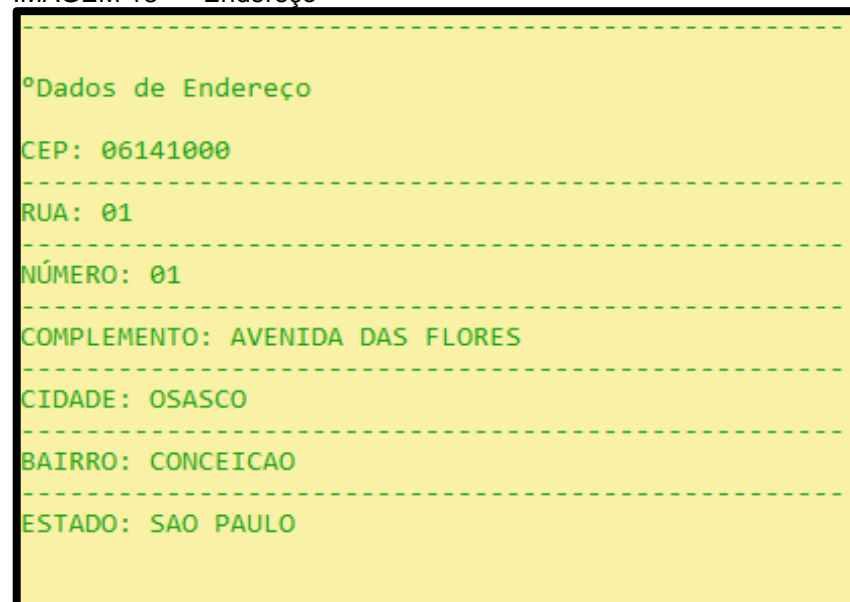
=>DIA: 23

=>MÊS: 02

=>ANO: 2020

FONTE: Autoria própria

IMAGEM 13 – Endereço

A yellow rectangular form with a black border. It contains several fields for address information, each preceded by a dashed green line. The fields are filled with green text: "ºDados de Endereço", "CEP: 06141000", "RUA: 01", "NÚMERO: 01", "COMPLEMENTO: AVENIDA DAS FLORES", "CIDADE: OSASCO", "BAIRRO: CONCEICAO", and "ESTADO: SAO PAULO".

ºDados de Endereço

CEP: 06141000

RUA: 01

NÚMERO: 01

COMPLEMENTO: AVENIDA DAS FLORES

CIDADE: OSASCO

BAIRRO: CONCEICAO

ESTADO: SAO PAULO

FONTE: Autoria própria

- Logo após o preenchimento dos dados de endereço o sistema irá perguntar se o paciente possui algum tipo de morbidade onde o funcionário deve informar qual é a morbidade que o paciente possui, caso contrário o sistema autopreenche esse campo como NENHUM.

IMAGEM 14 – Informar morbidade

```

-0 paciente possui algum tipo de morbidade? -
-[1]SIM -
-[2]NÃO -
-----
Escolha a opção desejada:

-----
-0 paciente possui algum tipo de morbidade? -
-[1]SIM -
-[2]NÃO -
-----
Escolha a opção desejada: 1
-----
MORBIDADE DO PACIENTE: Diabetes

```

FONTE: Autoria própria

3.3 Arquivos

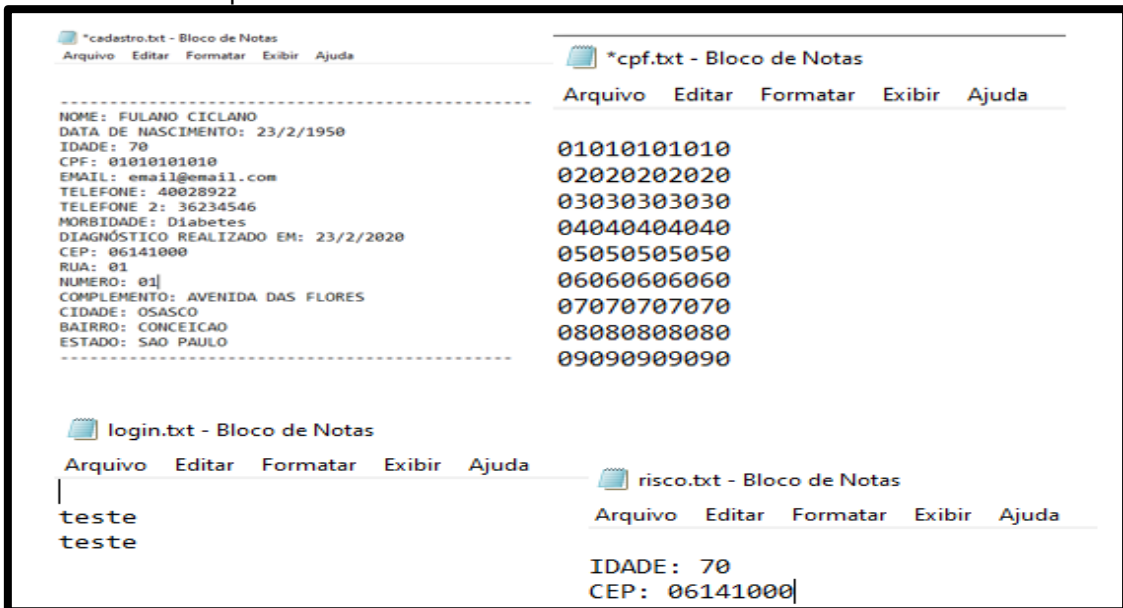
Todos os dados informados no sistema fica armazenado em um arquivo que funciona como uma espécie de banco de dados. É nesse local onde pode ser consultados todos os dados do paciente.

Após a primeira execução do sistema é criado 4 arquivos do tipo “txt” no local onde está salvo o código cadastro.txt para ser compilado e executado.

- O arquivo *cadastro.txt* é onde fica armazenado as informações de todos os pacientes cadastrados no sistema
- O arquivo *cpf.txt* é onde fica armazenado os CPF’S cadastrados no sistema
- O arquivo *login.txt* é onde fica armazenado o usuário e senha para entrar no sistema A primeira linha representa o login , já a segunda linha representa a senha, para fins de testes escolhemos como usuário e senha a palavra “teste”, más o funcionário pode cadastrar qualquer usuário e senha com letras, números ou símbolos.

- O arquivo *risco.txt* é onde fica armazenado as informações dos pacientes que fazem para do grupo de risco

IMAGEM 15 – Arquivos



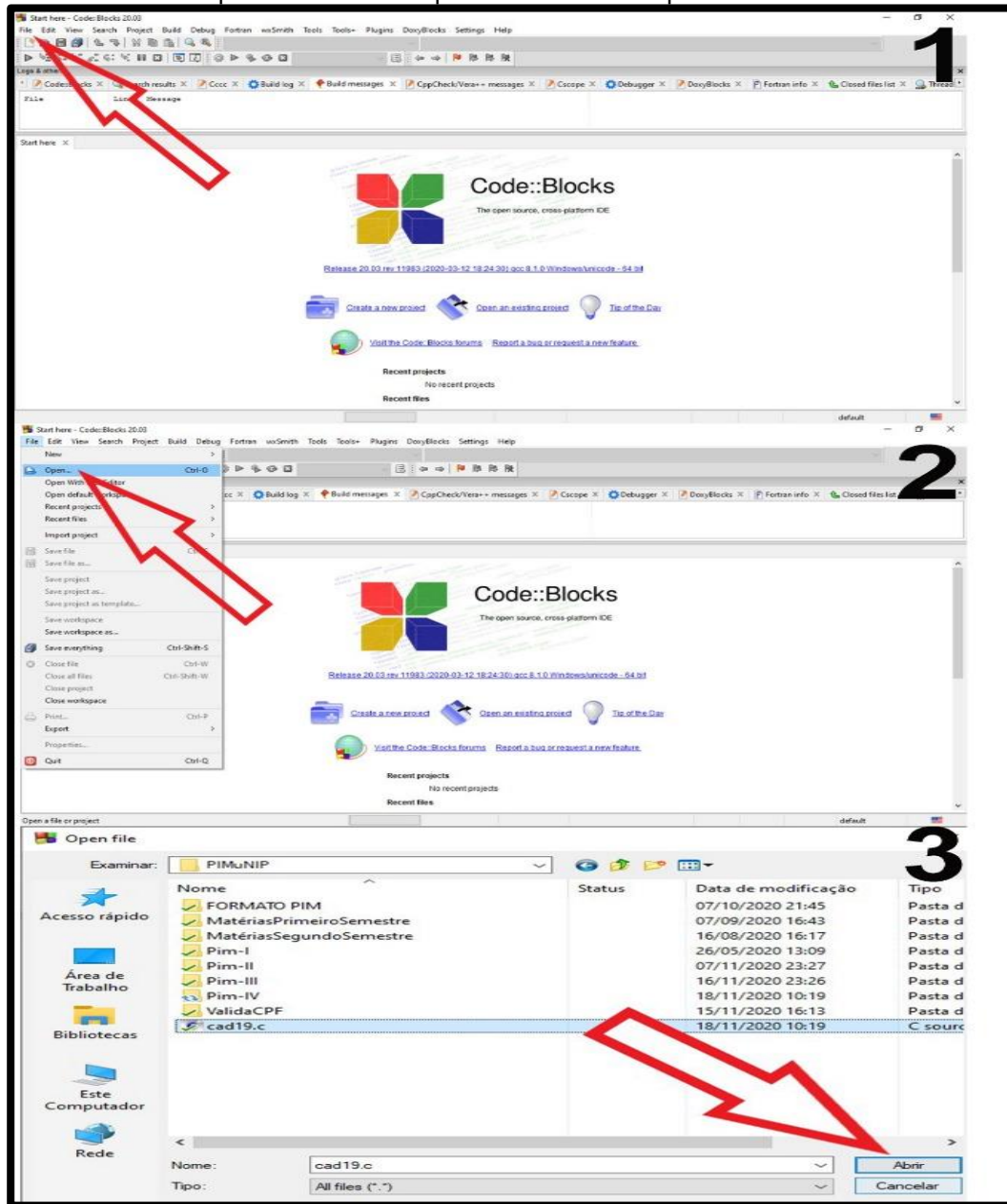
FONTE: Autoria própria

3.4 Execução do sistema

Para a execução do sistema o usuário deve ter alguma IDE(Ambiente de desenvolvimento integrado) instalado no computador onde será utilizado o sistema, Com o programa instalado o usuário deve abrir o arquivo cad19.c para compilar e executar. Abaixo segue a instrução passo-a-passo.

Com a IDE aberta(como exemplo estamos utilizando o code blocks) o usuário dedeve clicar em FILE, clicar em OPEN, escolher o local onde está salvo o cad19.c, selecionar o arquivo e clicar em ABRIR.

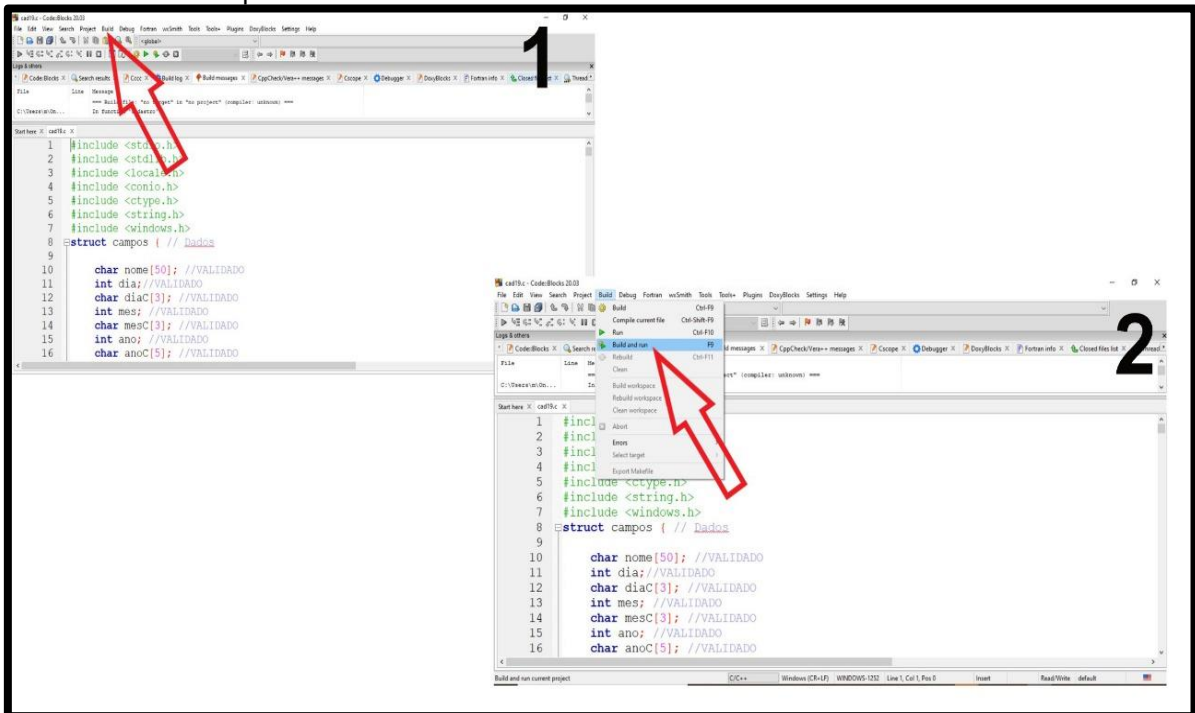
IMAGEM 16 – Preparando o sistema para executar e compilar



FONTE: Autoria própria

- Após o processo anterior o usuário deve compilar e executar o programa, para isso é só clicar em BUILD e em seguida em BUILD AND RUN.

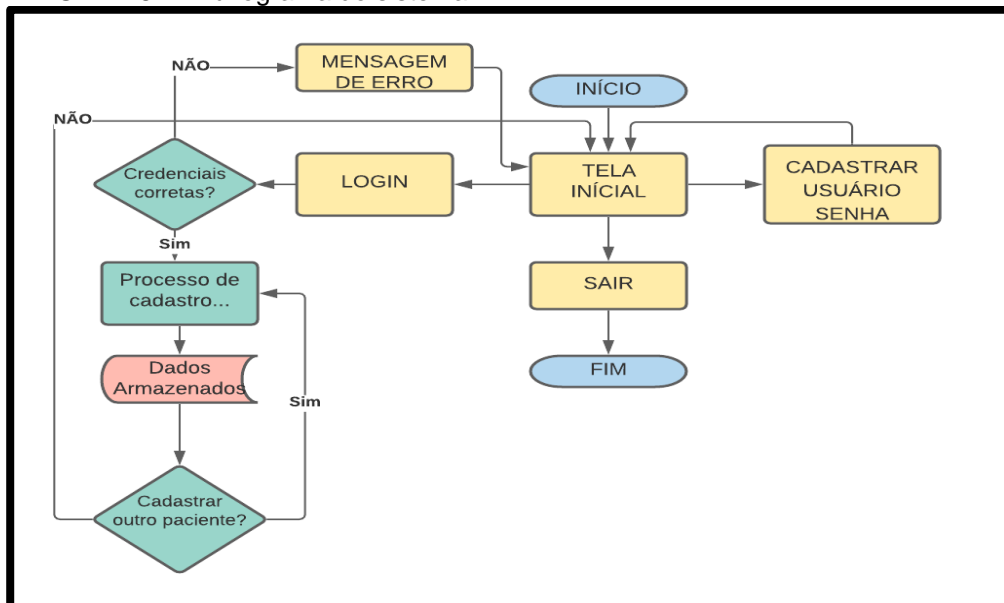
IMAGEM 17 – Compilando e executando o sistema



FONTE: Autoria própria

3.5 Fluxograma

IMAGEM 18 – Fluxograma do sistema



FONTE: Autoria própria

4 CONCLUSÃO

De acordo com as matérias fornecidas como base de estudo e pesquisa, foi possível a criação de um sistema programado em Linguagem C, para ambientes hospitalares, com o propósito de auxiliar profissionais de saúde na execução de cadastrar pacientes que apresentam quadros de contaminação por covid-19 e com uso destes dados clínicos podendo compartilhar dados com a secretária de saúde na luta e prevenção contra o vírus.

Para a concepção do sistema utilizamos os conceitos explorados, na disciplina Linguagem e Técnicas de Programação, fizemos uso do Ide e do Code Blocks para execução das linhas de códigos necessários para a criação dos campos que o sistema deve possuir, começando da tela de login e finalizando na pergunta se o usuário deseja realizar um novo cadastro.

Utilizando os conceitos explorados ao longo das aulas de Engenharia de Software I, entre todas as metodologias avaliamos seus pontos fortes e fracos, optamos por adotar a Metodologia Cascata, devido a mesma estar em alinhamento com nosso projeto e o desenvolvimento ser efetivado em etapas.

Com o aumento de casos de infecção por covid-19 e pouco conhecimento sobre o vírus, podemos utilizar a tecnologia para a captação de dados dos pacientes e ajudar os profissionais da saúde em um momento tão delicado em que o mundo atual se encontra, facilitando e agilizando o atendimento em hospitais.

O projeto se tornou viável e nos testes o sistema se demonstrou promissor, confiável e estável.

REFERÊNCIAS

ANÁLISE DE REQUISITOS. **O que são Requisitos Funcionais e Não Funcionais?**. Disponível em: <https://analisederequisitos.com.br/requisitos-funcionais-e-nao-funcionais/>. Acesso em: 19 nov. 2020.

DICAS DE PROGRAMAÇÃO. **O que são Funções e Procedimentos?**. Disponível em: [https://dicasdeprogramacao.com.br/o-que-sao-funcoes-e-procedimentos/O que são Funções e Procedimentos?](https://dicasdeprogramacao.com.br/o-que-sao-funcoes-e-procedimentos/O%20que%20sao%20Funcoes%20e%20Procedimentos?). Acesso em: 20 nov. 2020.

DICIO. **Requisitos**. Disponível em: <https://www.dicio.com.br/requisitos/>. Acesso em: 19 nov. 2020.

MEDIUM. **O Modelo em Cascata**. Disponível em: <https://medium.com/contexto-delimitado/o-modelo-em-cascata-f2418addaf36>. Acesso em: 20 nov. 2020.

MONITORA. **As principais metodologias de desenvolvimento de software que você precisa saber**. Disponível em: <https://www.monitoretec.com.br/blog/metodologias-de-desenvolvimento-de-software/#:~:text=As%20metodologias%20de%20desenvolvimento%20de,escolha%20da%20metodologia%20mais%20adequada>. Acesso em: 19 nov. 2020.

WIKIPÉDIA. **C (linguagem de programação)**. Disponível em: [https://pt.wikipedia.org/wiki/C_\(linguagem_de_programa%C3%A7%C3%A3o\)](https://pt.wikipedia.org/wiki/C_(linguagem_de_programa%C3%A7%C3%A3o)). Acesso em: 19 nov. 2020.

WIKIPÉDIA. **Code::Blocks**. Disponível em: <https://pt.wikipedia.org/wiki/Code::Blocks>. Acesso em: 18 nov. 2020.

WIKIPÉDIA. **Engenharia de software**. Disponível em: https://pt.wikipedia.org/wiki/Engenharia_de_software. Acesso em: 19 nov. 2020.