

**UNIVERSIDADE TECNOLÓGICA FEDERAL DO
PARANÁ BACHARELADO EM ENGENHARIA DA
COMPUTAÇÃO RELATÓRIO TÉCNICO -
PROGRAMAÇÃO ORIENTADA A OBJETOS**

**ANA CAROLINA RIBEIRO MIRANDA
DARLAN OLIVEIRA SANTOS
GUSTAVO RODRIGUES BASSACO
LUCAS EDUARDO PIRES PARRA
RODRIGO OLIVEIRA BADEGA**

**INCLUSÃO DOS DEFICIENTES VISUAIS NO CADASTRO
NACIONAL DE INFORMAÇÕES SOCIAIS**

APUCARANA, 2021

**ANA CAROLINA RIBEIRO MIRANDA
DARLAN OLIVEIRA SANTOS
GUSTAVO RODRIGUES BASSACO
LUCAS EDUARDO PIRES PARRA
RODRIGO OLIVEIRA BADEGA**

INCLUSÃO DOS DEFICIENTES VISUAIS NO CADASTRO NACIONAL DE INFORMAÇÕES SOCIAIS

Relatório Técnico do Trabalho Multidisciplinar (TM) apresentado como requisito parcial para obtenção de créditos na disciplina de Programação Orientada a Objetos da Universidade Tecnológica Federal do Paraná.

Orientador: Prof. Dr. Lucio Agostinho Rocha

APUCARANA, 2021

RESUMO

Este presente relatório apresenta uma proposta de incluir deficientes visuais no cadastro nacional de informações sociais (CNIS) a fim de garantir a melhor visualização podendo ajustar o tamanho da fonte e o tema da tela a fim de auxiliar pessoas com deficiência visual parcial e/ou fotossensibilidade. Este programa visa monitorar os acessos por meio de características como a herança, o polimorfismo e o tratamento de exceções, tudo a partir de uma interface amigável e interativa. A metodologia propõe o desenvolvimento em linguagem de programação JAVA, através da plataforma de IDE denominada Eclipse. O resultado obtido foi um programa que pode se ajustar a fonte, mudar para o tema escuro, fazer o login e por fim a tela de cadastro.

Palavras-chave: Inclusão. Deficiência Visual. CNIS. Tecnologia. Informações.

LISTA DE ILUSTRAÇÕES

Figura 1 - Declaração das Classes e suas Heranças.....	8
Figura 2 - Declaração das Classes e suas Heranças.....	8
Figura 3 - Declaração das Classes e suas Heranças.....	9
Figura 4 - Declaração Variáveis.....	10
Figura 5 - Função mudaLayout.....	10
Figura 6 - Tratamento de Exceções TelaCadastro.....	11
Figura 7 - Tela de Login com informações inseridas.....	12
Figura 8 - Tela de Cadastro no modo Grande e Escuro.....	12
Figura 9 - Tela de Exibe Cadastro no modo Grande e Escuro com informações inseridas e salvas.....	13
Figura 10 - Saída Obtida com a Execução do Programa.....	14
Figura 11 - Tela de Login com informações inseridas.....	15
Figura 12 - Tela de Login com informações erradas inseridas.....	15
Figura 13 - Tela de Cadastro no modo Grande e Escuro com informações inseridas.....	16
Figura 14 - Tela de Cadastro no modo Grande e Escuro com informações não inseridas.....	16
Figura 15 - Tela de Exibe Cadastro no modo Grande e Escuro com informações inseridas e salvas.....	17
Figura 16 - Diagrama UML do Projeto.....	18

SUMÁRIO

1	INTRODUÇÃO.....	6
2	TRABALHOS RELACIONADOS.....	7
3	DESENVOLVIMENTO.....	8
	3.1 Utilização de Herança.....	8
	3.2 Utilização de Polimorfismo.....	8
	3.3 Utilização Tratamento de Exceções.....	10
	3.4 Utilização da Interface Gráfica com o Usuário (GUI).....	11
	3.5 Link Público do Código-Fonte.....	13
4	RESULTADOS.....	14
5	CONSIDERAÇÕES FINAIS.....	20
6	REFERÊNCIAS.....	21

1 INTRODUÇÃO

A visão é o canal de extrema importância para os relacionamentos com os indivíduos do mundo. Estudos recentes revelam que enxergar não é uma habilidade inata, é preciso aprender a ver, não só conscientemente, o desenvolvimento das funções visuais ocorrem nos primeiros anos de vida, até recentemente, não se levava em conta a existência de resíduos visuais, as pessoas eram tratadas como se fossem cegas, aprendendo a ler e escrever em braille e utilizar as bengalas. Hoje em dia estudos buscam o desenvolvimento e melhoramento na vida cotidiana e no lazer, desenvolvendo técnicas significativamente a qualidade de vida, mesmo sem eliminar a deficiência, utilizando meios que auxiliam opticamente pessoas com baixa visão ou com fotossensibilidade.

Com isso, realiza-se a seguinte pergunta: “É possível desenvolver um sistema que seja acessível às pessoas com deficiência visual?”. O Projeto proposto tem como objetivo a criação de um programa de cadastro do governo CNIS, (Cadastro Nacional de Informações Sociais), inclusivo para pessoas com deficiência visual, que, segundo a secretaria da justiça, é o comprometimento parcial de 40% a 60% ou total da visão, não sendo considerados deficientes visuais pessoas com doenças como miopia, astigmatismo ou hipermetropia que podem ser corrigidas com o uso de lentes ou em cirurgias. (AMPUDIA, Ricardo).

O cadastro nacional de informações sociais, que possui a sigla CNIS é um documento utilizado pelo Instituto Nacional do Seguro Social (INSS) para saber se um cidadão já pode conseguir sua aposentadoria, o CNIS é um banco de informações utilizado pelo governo federal para reunir dados sobre contribuições previdenciárias e informações trabalhistas de um cidadão durante toda sua vida profissional, informações reunidas utilizadas para definir quando e com qual remuneração o cidadão poderá se aposentar. (Ribeiro, Matheus Vinicius).

O restante deste relatório é apresentado como segue. A Seção 2 apresenta os trabalhos relacionados que foram pesquisados. A Seção 3 apresenta o desenvolvimento do projeto proposto; a Seção 4 apresenta os resultados; a Seção 5 faz as considerações finais.

2 TRABALHOS RELACIONADOS

Com o avanço de informações e comunicação em expansão no mundo, a ciência vem procurando aumentar ao máximo o aproveitamento da tecnologia, que é uma das principais formas de comunicação, ferramenta essencial no mercado de trabalho, grande parcela da comunidade ainda não está inserida neste contexto, como é o caso dos deficientes visuais. Com o intuito de garantir os direitos de pessoas com deficiência visual e incorporá-los nesta sociedade informatizada, através de sistemas computacionais planejados para essas pessoas.

Como o projeto proposto em questão é uma forma inovadora de inclusão digital para pessoas com deficiência visual parcial, para realizarem o cadastro nacional de informações sociais (CNIS), não existem trabalhos semelhantes no ramo.

3 DESENVOLVIMENTO

Na criação do programa em questão, utilizamos a linguagem de programação JAVA, através do ambiente de programação de IDE chamada Eclipse, realizados no sistema operacional Windows 10.

A linguagem JAVA possui como principais vantagens são a portabilidade, que permite que a aplicação execute em praticamente qualquer sistema, sua robustez garante que a aplicação não interfira em outras atividades do Sistema Operacional, além de possuir uma segurança forte contra ataques não-intencionais, e sua orientação a objetos permite diversas vantagens de reutilização de código desse paradigma, como herança e polimorfismo.

3.1 Utilização de Herança

A utilização de Herança se dá em quase todas as classes, em principal as classes que possuem interface gráfica como por exemplo a TelaLogin, TelaCadastro e ExibeCadastro, os quais possuem herança do JFrame.

Figura 1, 2 e 3 - Declaração das Classes e suas Heranças

```

9
10 public class ExibeCadastro extends javax.swing.JFrame {
11
12     private Variaveis var = new Variaveis();
13     private javax.swing.JButton jbMCor;

```

```

9
10 public class TelaCadastro extends javax.swing.JFrame {
11
12     private Variaveis var = new Variaveis();
13     private Filho filho = new Filho();
14     private Mae mae = new Mae();
15     private Pai pai = new Pai();
16     private javax.swing.JButton jbMCor;

```



```

9 public class TelaLogin extends javax.swing.JFrame {
10
11     private Variaveis var = new Variaveis();
12     private javax.swing.JButton jbEntrar;

```

Fonte: Autoria Própria (2021).

Descrição detalhada do código-fonte do trecho apresentado a figura 1, 2 e 3:

Linha 10: A classe declarada herda as funções do JFrame.

Linha 11: Criando uma variavel var do tipo Variaveis.

3.2 Utilização de Polimorfismo

O polimorfismo se dá na classe Variaveis, onde é criada uma lista dinâmica da classe Pessoa que recebe variáveis da subclasses Filho, Pai e Mae.

Figura 4 - Declaração Variáveis

```

5 public class Variaveis {
6
7     private boolean modoGrande = false;
8     private boolean modoEscuro = false;
9     private ArrayList<Pessoa> lista = new ArrayList<>();
10
11     public void inserir(Filho filho){
12         this.lista.add(filho);
13     }

```

Fonte: Autoria Própria (2021).

Descrição detalhada do código-fonte do trecho apresentado:

Linha 5: Declarando classe Variaveis.

Linha 9: Criando um ArrayList da classe pessoa.

Linha 12: Adicionando o componente filho na lista

3.3 Utilização Tratamento de Exceções

Alguns tratamentos de exceções presentes na classe de TelaLogin e TelaCadastro, ao qual irão fazer as validações respectivas para mudar o layout dos JFrame e para inserção dos dados na tela de cadastro.

Figura 5 - Função mudaLayout

```

271     public void mudalayout(){
272
273         try {
274             for (javax.swing.UIManager.LookAndFeelInfo info : javax.swing.UIManager.getInstalledLookAndFeels()) {
275                 if ("Nimbus".equals(info.getName())) {
276                     javax.swing.UIManager.setLookAndFeel(info.getClassName());
277                     break;
278                 }
279             }
280         } catch (ClassNotFoundException | InstantiationException | IllegalAccessException | javax.swing.UnsupportedLookAndFeelException ex) {
281             java.util.logging.Logger.getLogger(TelaLogin.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);
282         }
283     } //mudalayout
284
285 } //TelaLogin

```

Fonte: Autoria Própria (2021).

Ao se criar um projeto novo no Netbeans, tem-se como padrão o look and feel “Nimbus” da Oracle, esse look and feel torna a visão da janela diferente do que é projetado na IDE, para que fique padrão ao que se desenha.

Figura 6 - Tratamento de Exceções TelaCadastro

```

108     jbConcluir.setText("Concluir");
109     jbConcluir.addActionListener(new java.awt.event.ActionListener() { //adiciona ao botao uma funcao
110         public void actionPerformed(java.awt.event.ActionEvent evt) {
111             try { //testa se tem campos nulos
112                 jbConcluirActionPerformed(evt); //chama metodo de concluir cadastro
113             } catch (NullPointerException e) { //se tiver
114                 e.printStackTrace(); //printa stack
115                 jlWarning.setVisible(true); //deixa warning visivel
116             } //catch
117         }
118     });

```

Fonte: Autoria Própria (2021).

Descrição detalhada do código-fonte do trecho apresentado:

Linha 111: Declarando tratamento de exceções.

Linha 112: Chama método de concluir cadastro

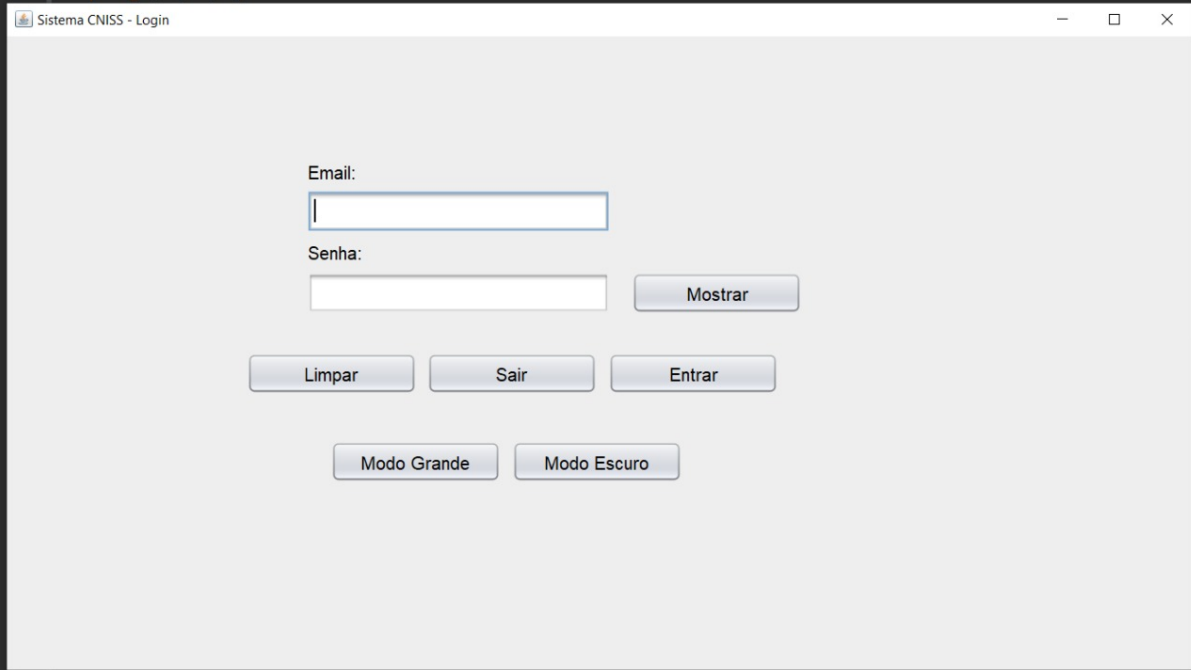
Linha 113: Caso exista algum campo não preenchido, no caso null, irá deixar um warning visível e printar o stack.

3.4 Utilização da Interface Gráfica com o Usuário (GUI)

Possuem no total três telas com interface gráfica com o usuário, sendo elas as telas Login, TelaCadastro e ExibeCadastro, ao qual irão pedir as informações respectivamente para-se acessar o sistema, realizar o cadastro inserindo as

informações necessárias e por fim uma tela que irá exibir os dados cadastrados em questão.

Figura 7 - Tela de Login com informações inseridas.



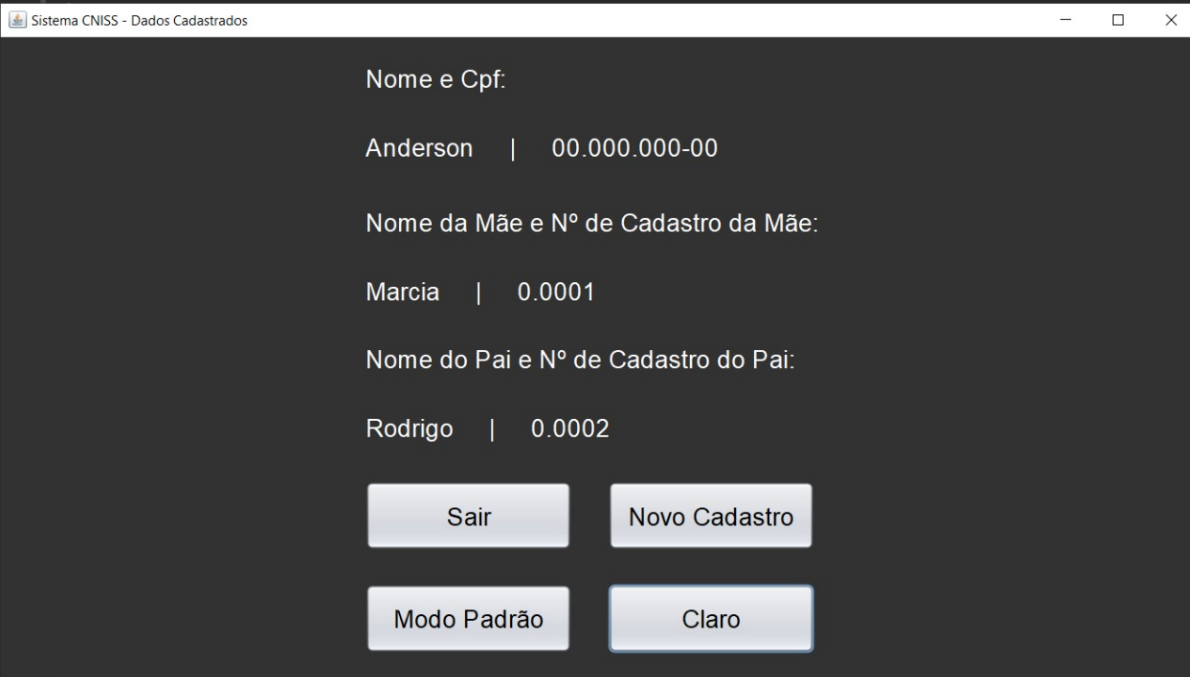
Fonte: Autoria Própria (2021).

Figura 8 - Tela de Cadastro no modo Grande e Escuro



Fonte: Autoria Própria (2021).

Figura 9 - Tela de Exibe Cadastro no modo Grande e Escuro com informações inseridas e salvas.



Sistema CNISS - Dados Cadastrados

Nome e Cpf:

Anderson | 00.000.000-00

Nome da Mãe e N° de Cadastro da Mãe:

Marcia | 0.0001

Nome do Pai e N° de Cadastro do Pai:

Rodrigo | 0.0002

Sair Novo Cadastro

Modo Padrão Claro

Fonte: Autoria Própria (2021).

3.5 Link Público do Código-Fonte

O código fonte pode ser encontrado pelo repositório on-line com acesso público Github, pelo Link:

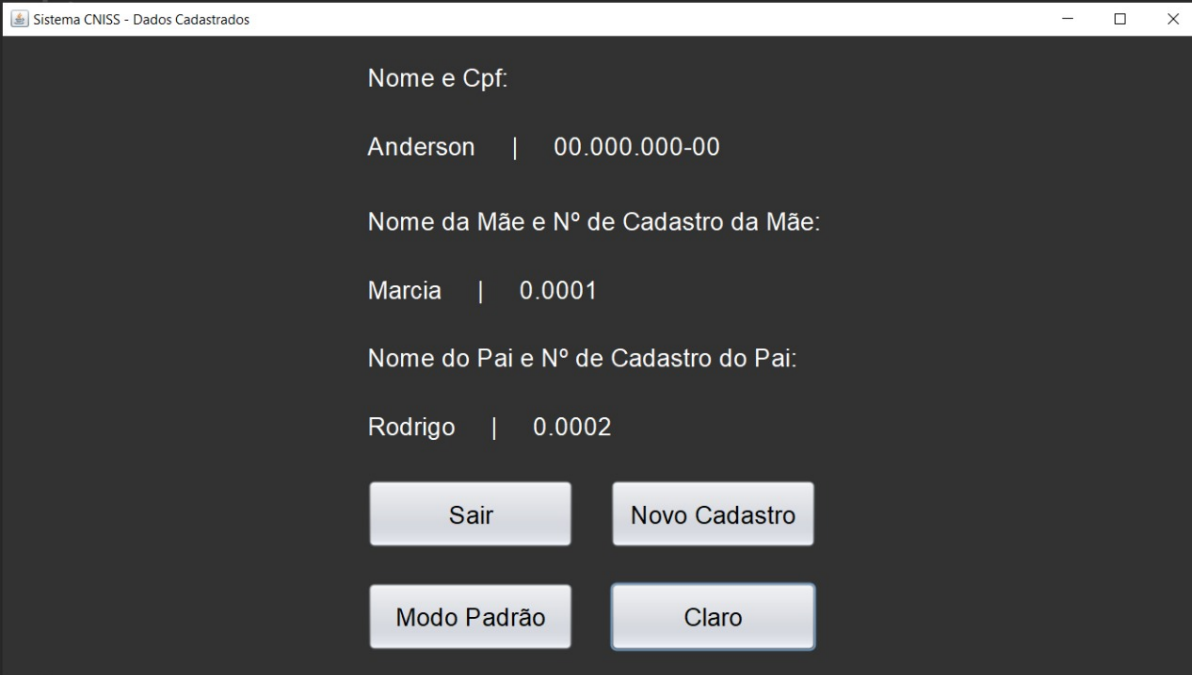
- <https://github.com/Aninha33/Trabalho-Multidisciplinar-POO.git>

4 RESULTADOS

4.1 Resultados da Utilização de funções com passagem de parâmetros por referência

A saída final obtida com a execução do programa através de funções com passagem de parâmetros por referência se encontra na última tela ExibeCadastro, ao qual contém as informações que foram inseridas na tela de cadastro, no entanto, existem outras classes ao quais também contém funções de passagem de parâmetros por referências tais como TelaCadastro.

Figura 10 - Saída Obtida com a Execução do Programa.



Sistema CNISS - Dados Cadastrados

Nome e Cpf:

Anderson | 00.000.000-00

Nome da Mãe e Nº de Cadastro da Mãe:

Marcia | 0.0001

Nome do Pai e Nº de Cadastro do Pai:

Rodrigo | 0.0002

Sair Novo Cadastro

Modo Padrão Claro

Fonte: Autoria Própria (2021).

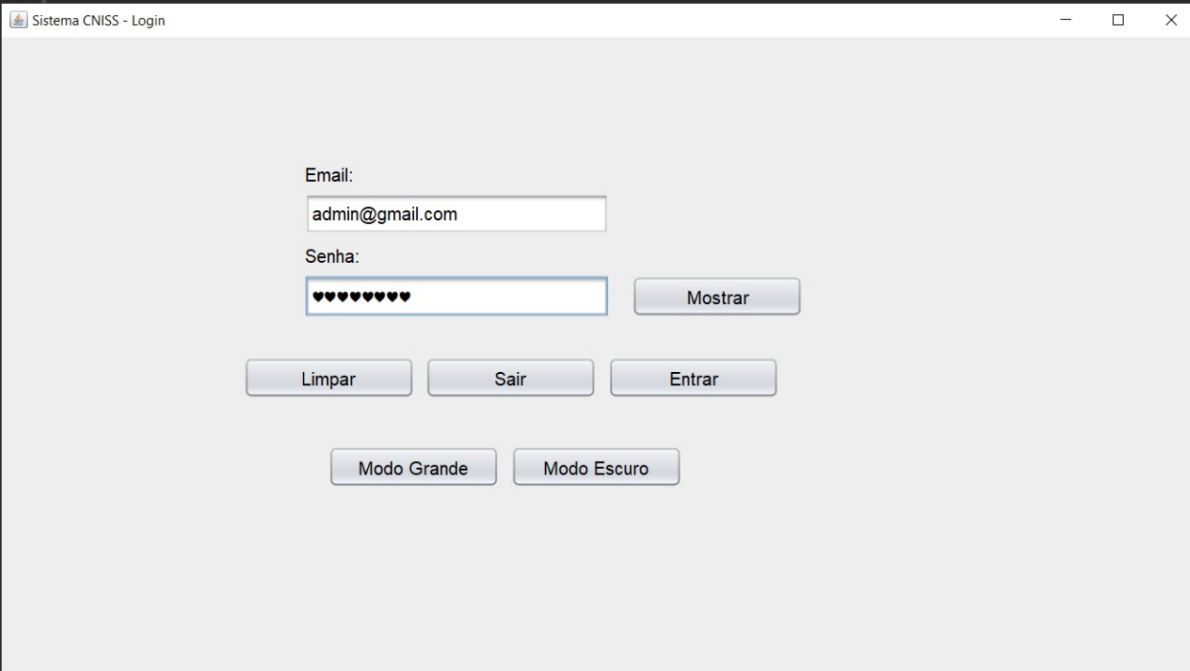
4.2 Resultados da Utilização de Herança e de Polimorfismo

A utilização da herança nas classes do programa são funcionais e não apresentaram problemas, visto que a lista de resultados foi percorrida corretamente apresentando as mensagens e funcionalidades desejadas. Outra funcionalidade apresentada sem problemas foi o ArrayList salvando polimorficamente na classe TelaCadastro que ao ser concluída, é impressa na classe ExibeCadastro.

4.4 Resultados da Interface Gráfica com o Usuário (GUI) e do Tratamento de Exceções

A utilização dos tratamentos de exceções implementadas juntamente com a interface gráfica não apresentaram problemas, podemos visualizar seus respectivos resultados ao serem inseridas as informações e caso não estejam conforme o esperado.

Figura 11 - Tela de Login com informações inseridas.



Sistema CNISS - Login

Email:

Senha:

Fonte: Autoria Própria (2021).

Figura 12 - Tela de Login com informações erradas inseridas

The screenshot shows a web application window titled "Sistema CNISS - Login". It features a login form with the following elements:

- Email:** A text input field containing "aaaa".
- Senha:** A password input field with four heart icons, indicating it is masked. To its right is a "Mostrar" button.
- Error Message:** Below the password field, a red text message reads "Email ou senha incorretos!".
- Buttons:** Below the error message are three buttons: "Limpar", "Sair", and "Entrar".
- Theme Selection:** At the bottom of the form are two buttons: "Modo Grande" and "Modo Escuro".

Fonte: Autoria Própria (2021).

Figura 13 - Tela de Cadastro no modo Grande e Escuro com informações inseridas.

The screenshot shows a web application window titled "Sistema CNISS - Cadastro" in dark mode. It features a registration form with the following elements:

- Nome:** A text input field containing "Anderson".
- CPF:** A text input field containing "00.000.000-00".
- Nome da Mãe:** A text input field containing "Marcia".
- Nº de Cadastro da Mãe:** A text input field containing "0.0001".
- Nome do Pai:** A text input field containing "Rodrigo".
- Nº de Cadastro do Pai:** A text input field containing "0.0002".
- Buttons:** At the bottom of the form are four buttons: "Sair", "Concluir", "Modo Padrão", and "Claro".

Fonte: Autoria Própria (2021).

Figura 14 - Tela de Cadastro no modo Grande e Escuro com informações não inseridas.

Sistema CNISS - Cadastro

Nome:

CPF:

Nome da Mãe:

Nº de Cadastro da Mãe:

Nome do Pai:

Nº de Cadastro do Pai:

Preencha todos os campos!

Sair Concluir

Modo Padrão Claro

Fonte: Autoria Própria (2021).

Figura 15 - Tela de Exibe Cadastro no modo Grande e Escuro com informações inseridas e salvas.

Sistema CNISS - Dados Cadastrados

Nome e Cpf:

Anderson | 00.000.000-00

Nome da Mãe e Nº de Cadastro da Mãe:

Marcia | 0.0001

Nome do Pai e Nº de Cadastro do Pai:

Rodrigo | 0.0002

Sair Novo Cadastro

Modo Padrão Claro

Fonte: Autoria Própria (2021).

4.6 Diagrama UML do projeto

Inicialmente, tem-se que o usuário irá se comunicar com a tela de Login em questão, onde poderá ter acesso às configurações, podendo editar para o formato de fonte maior ou ao modo escuro. Logo após logar, acessa-se a tela de Cadastro, ao qual irá preencher os requisitos solicitados no diagrama que podemos visualizar na Figura 8, a tela em questão também tem acesso às configurações. O diagrama UML de classe do diagrama do projeto proposto possui as seguintes características:

Para melhor visualização, é possível acessar o esquema completo realizado pelo site <https://www.smartdraw.com> conhecido como Smart Draw com o seguinte Link abaixo:

- <https://cloud.smartdraw.com/share.aspx/?pubDocShare=A2510154EADCAED760BD89B4FFA716F4A21>

5 CONSIDERAÇÕES FINAIS

Nessa pesquisa foi implementado um programa de criação de cadastro nacional de informações sociais inclusivo para pessoas com deficiência visual parcial e/ou fotossensibilidade, utilizando a linguagem JAVA com o conhecimento adquirido pela matéria de Programação Orientada a Objetos.

Obtendo-se um bom desempenho no programa, sem apresentação de erros, pondo em prática os conhecimentos esperados, resultando em um enriquecimento aos desenvolvedores.

REFERÊNCIAS

- [1] AMPUDIA, Ricardo. “O que é deficiência visual?”. Disponível em <<https://novaescola.org.br/conteudo/270/deficiencia-visual-inclusao>>. Acesso em 27 de Novembro de 2021.
- [2] GIL, Marta. “Deficiência Visual”. Disponível em: <<http://portal.mec.gov.br/seed/arquivos/pdf/deficienciavisual.pdf>>. Acesso em 27 de Novembro de 2021.
- [3] MELO, Simone Azevedo Bandeira; SILVA, Joabe Souza; ARAÚJO, Laryssa Nogueira dos Anjos. “Vision Access: Acessibilidade para Deficientes Visuais Utilizando Reconhecimento e Síntese de Voz”. Disponível em: <<https://propi.ifto.edu.br/ocs/index.php/connepi/vii/paper/viewFile/1927/2975>> . Acesso em 27 de Novembro de 2021.
- [4] RIBEIRO, Matheus Vinicius. “CNIS: Aprenda como resolver os problemas os problemas deste sistema”. Disponível em: <<https://www.jornalcontabil.com.br/cnis-aprenda-como-resolver-os-problemas-deste-sistema/>> . Acesso em 27 de Novembro de 2021.