**CENTRO ESTADUAL DE EDUCAÇÃO TECNOLÓGICA CENTRO PAULA SOUZA**

**ETEC DA ZONA LESTE**

**Ensino Médio com Habilitação Profissional de Técnico em Desenvolvimento de Sistemas**

**João Vitor de Sousa Rodrigues**

**Lucas Pereira Carvalho**

**Paulo Enrick Santos Silva**

**Tamiris Romano de Carvalho**

**SCANNERRF: Sistema de identificação de pessoas com reconhecimento facial**

**São Paulo**

**2023**

**João Vitor de Sousa Rodrigues**

**Lucas Pereira Carvalho**

**Paulo Enrick Santos Silva**

**Tamiris Romano de Carvalho**

# SCANNERRF: Sistema de identificação de pessoas com reconhecimento facial

Trabalho de Conclusão de curso apresentado ao Curso Técnico em Desenvolvimento de Sistemas da Etec Zona Leste, orientado pelo Prof. Jeferson Roberto de Lima, como requisito parcial para obtenção do título de técnico em Desenvolvimento de Sistemas.

**São Paulo**

**2023**

Dedicamos este trabalho aos nossos familiares, professores e colegas de classe, que nos auxiliaram a terminar esse projeto com êxito e caminharam conosco durante esses 3 anos.

**AGRADECIMENTOS**

Expressamos nossa gratidão a todos os professores por trabalharem com profissionalismo e respeito, especialmente aos nossos coordenadores Jeferson Roberto de Lima e Rogério Bezerra Costa.

“A primeira regra de qualquer tecnologia utilizada nos negócios é que a automação aplicada a uma operação eficiente aumentará a eficiência. A segunda é que a automação aplicada a uma operação ineficiente aumentará a ineficiência”.

Bill Gates

**RESUMO**

A falta de segurança nas escolas tem sido um problema preocupante que afeta a sociedade como um todo, vemos isso por conta das várias ondas de violência que acontecem. Uma solução viável para esse problema é desenvolver um sistema para identificação e monitoramento em um ambiente escolar utilizando a tecnologia de reconhecimento facial, com intuito de fiscalizar a entrada dos alunos a fim de evitar possíveis invasões. A metodologia do projeto é baseada em cinco etapas que consistem na execução e finalização do trabalho com diferentes métodos. Aborda combinações qualitativas e quantitativas, aproveitando assim diferentes tipos de dados para as pesquisas e o objetivo do trabalho.

**Palavras-chave**: Segurança. Violência. Reconhecimento facial.

**ABSTRACT**

The lack of security in schools has been a worrying problem that affects society in general, it is possible to see this because of the recent wave of violence that has been happening. A viable solution to this problem is to develop a system for identification and monitoring in a school environment using facial recognition technology, in order to supervise the entry of students to avoid possible invasions. The project methodology is based on five steps that consist of executing and finalizing the work with different methods. It addresses qualitative and quantitative combinations, thus taking advantage of different types of data for research and the objective of the work.

**Keywords**: Security. Violence. Facial recognition.

**LISTA DE FIGURAS**

**FIGURAS:**

[Figura 1: Exemplo de formulário de cadastro em Python 18](#_Toc143097143)

[Figura 2: Resultado do formulário de cadastro em Python 20](#_Toc143097144)

[Figura 3: Estilizando o código com PyQt5 21](#_Toc143097145)

[Figura 4: Continuação da estilização do código 22](#_Toc143097146)

[Figura 5: Resultado da estilização do código com PyQt5 25](#_Toc143097147)

[Figura 6: Exemplo de código para detecção por reconhecimento facial utilizando o OpenCV ............26](#_Toc143097148)

[Figura 7: Diagrama de Entidade-Relacionamento 28](#_Toc143097149)

[Figura 8: Modelo de Entidade e Relacionamento 29](#_Toc143097150)

[Figura 9: Diagrama de Caso de Uso - ScannerRF 30](#_Toc143097151)

[Figura 10: Diagrama de Classe - ScannerRF 42](#_Toc143097152)

[Figura 11: Diagrama de Sequência - Atualizar Aluno 43](#_Toc143097153)

[Figura 12: Diagrama de Sequência - Cadastro do Aluno 43](#_Toc143097154)

[Figura 13: Diagrama de Sequência - Login 44](#_Toc143097155)

[Figura 14: Diagrama de Sequência - Reconhecer Aluno 45](#_Toc143097156)

[Figura 15: Diagrama de Sequência do Visualizar Aluno 46](#_Toc143097157)

[Figura 16: Diagrama de Sequência do Visualizar Entrada 47](#_Toc143097158)

[Figura 17: Diagrama de Atividade Login 48](#_Toc143097159)

[Figura 18: Diagrama de Atividade Cadastro do Aluno 49](#_Toc143097160)

[Figura 19: Diagrama de Atividade Reconhecer Face 50](#_Toc143097161)

[Figura 20: Diagrama de Atividade Visualizar Entrada 51](#_Toc143097162)

[Figura 21: Diagrama de Atividade Visualizar Turma 51](#_Toc143097163)

[Figura 22: Diagrama de Atividade: Visualizar Aluno 52](#_Toc143097164)

[Figura 23: Diagrama de Atividade Pesquisar Entrada 52](#_Toc143097165)

[Figura 24: Wireframe de Baixa Fidelidade – Login 53](#_Toc143097166)

[Figura 25: Wireframe de Baixa Fidelidade – Página Inicial (Diretor acadêmico) 54](#_Toc143097167)

[Figura 26: Wireframe de Baixa Fidelidade – Página Inicial (Segurança) 55](#_Toc143097168)

[Figura 27: Wireframe de Baixa Fidelidade – Alunos (Diretor acadêmico) 56](#_Toc143097169)

[Figura 28: Wireframe de Baixa Fidelidade – Cadastrar Aluno (Diretor acadêmico) 57](#_Toc143097170)

[Figura 29: Wireframe de Baixa Fidelidade – Pesquisar Aluno (Diretor acadêmico) 58](#_Toc143097171)

[Figura 30: Wireframe de Baixa Fidelidade – Visualizar Aluno (Diretor acadêmico) 59](#_Toc143097172)

[Figura 31: Wireframe de Baixa Fidelidade – Visualizar Aluno (Diretor acadêmico) 60](#_Toc143097173)

[Figura 32: Wireframe de Baixa Fidelidade – Atualizar Aluno (Diretor acadêmico) 61](#_Toc143097174)

[Figura 33: Wireframe de Baixa Fidelidade – Histórico de Entradas (Diretor acadêmico) 62](#_Toc143097175)

[Figura 34: Wireframe de Baixa Fidelidade – Histórico de Entradas (Segurança) 63](#_Toc143097176)

[Figura 35: Wireframe de Baixa Fidelidade – Scanner (Segurança) 64](#_Toc143097177)

[Figura 36: Prototipação - Login 65](#_Toc143097178)

[Figura 37: Prototipação - Página Inicial (Diretor acadêmico) 66](#_Toc143097179)

[Figura 38: Prototipação - Página Inicial (Segurança) 67](#_Toc143097180)

[Figura 39: Prototipação - Alunos (Diretor acadêmico) 68](#_Toc143097181)

[Figura 40: Prototipação - Cadastrar Aluno (Diretor acadêmico) 69](#_Toc143097182)

[Figura 41: Prototipação - Pesquisar Aluno (Diretor acadêmico) 70](#_Toc143097183)

[Figura 42: Prototipação - Visualizar Aluno (Diretor acadêmico) 71](#_Toc143097184)

[Figura 43: Prototipação - Visualizar Turma (Diretor acadêmico) 72](#_Toc143097185)

[Figura 44: Prototipação - Atualizar Aluno (Diretor acadêmico) 73](#_Toc143097186)

[Figura 45: Prototipação - Histórico de Entradas (Diretor acadêmico) 74](#_Toc143097187)

[Figura 46: Prototipação - Histórico de Entradas (Segurança) 75](#_Toc143097188)

[Figura 47: Prototipação – Scanner (Segurança) 76](#_Toc143097189)

**LISTA DE QUADROS**

**QUADROS:**

[Quadro 1 – Visão geral do caso de uso da aplicação a partir da perspectiva do diretor acadêmico 33](#_Toc20904)

[Quadro 2 - Visão geral do caso de uso da aplicação a partir da perspectiva do segurança 34](#_Toc17502)

[Quadro 3 – Descrição do caso de uso “Fazer login” (Diretor acadêmico) 35](#_Toc23840)

[Quadro 4 – Descrição do caso de uso “Fazer login” (Segurança) 36](#_Toc16815)

[Quadro 5 – Descrição do caso de uso “Cadastrar aluno” (Diretor acadêmico) 37](#_Toc1198)

[Quadro 6 – Descrição do caso de uso “Visualizar turma” (Diretor acadêmico) 38](#_Toc9183)

[Quadro 7 – Descrição do caso de uso “Visualizar aluno” (Diretor acadêmico) 39](#_Toc12604)

[Quadro 8 – Descrição do caso de uso “Atualizar aluno” (Diretor acadêmico) 40](#_Toc17877)

[Quadro 9 – Descrição do caso de uso “Visualizar entrada” (Diretor acadêmico) 41](#_Toc5274)

[Quadro 10 - Descrição do caso de uso “Visualizar entrada” (Segurança) 42](#_Toc25720)

[Quadro 11 – Descrição do caso de uso “Pesquisar entrada” (Diretor acadêmico) 42](#_Toc13437)

[Quadro 12 – Descrição do caso de uso “Pesquisar entrada” (Segurança) 43](#_Toc8656)

[Quadro 13 - Descrição do caso de uso “Reconhecer face” (Segurança) 44](#_Toc9927)

**LISTA DE ABREVIATURAS E SIGLAS**

Cascading Style Sheet (CSS)

Computer Aided Software Engineering (CASE)

Diagrama de Entidade-Relacionamento (DER)

Extensible Markup Language (XML)

Graphical User Interface (GUI)

Hypertext Markup Language (HTML)

**Hypertext Preprocessor (PHP)**

Intergrated Development Environment (IDE)

Modelo de Entidade e Relacionamento (MER)

Open Source Computer Vision (OPENCV)

Registro de Matricula (RM)

Structured Query Language (SQL)

Unified Modeling Language (UML)

**Sumário**

[SCANNERRF: Sistema de identificação de pessoas com reconhecimento facial 12](#_Toc143089010)

[1. INTRODUÇÃO 11](#_Toc143089011)

[2. REFERENCIAL TEÓRICO 14](#_Toc143089012)

[2.1 A falta de segurança nas escolas 14](#_Toc143089013)

[2.1.1 Como a tecnologia pode ajudar a segurança nas escolas 14](#_Toc143089014)

[2.1.2 O reconhecimento facial no ambiente escolar 15](#_Toc143089015)

[2.2 Tecnologias Utilizadas 15](#_Toc143089016)

[2.2.1 UML 15](#_Toc143089017)

[2.2.2 Astah Community 16](#_Toc143089018)

[2.2.3 Visual Studio Code 17](#_Toc143089019)

[2.2.4 SQLITE3 17](#_Toc143089020)

[2.2.5 Python 17](#_Toc143089021)

[2.2.6 PyQt5 20](#_Toc143089022)

[2.2.7 OpenCV 25](#_Toc143089023)

[2.2.8 Figma 27](#_Toc143089024)

[3. DESENVOLVIMENTO 28](#_Toc143089025)

[3.1 Diagrama Entidade-Relacionamento (DER) 28](#_Toc143089026)

[3.1.1 Modelo Entidade-Relacionamento (MER) 29](#_Toc143089027)

[3.2 Diagrama de Caso de Uso 30](#_Toc143089028)

[3.2.1 Documentação dos Casos de Uso 30](#_Toc143089029)

[3.3 Diagrama de Classe 41](#_Toc143089030)

[3.5 Diagrama de Atividade 47](#_Toc143089031)

[3.6 Wireframe de Baixa Fidelidade 53](#_Toc143089032)

[3.7 Prototipação das Telas 65](#_Toc143089033)

[4. CONSIDERAÇÕES FINAIS 77](#_Toc143089034)

[REFERÊNCIAS 78](#_Toc143089035)

# 1. INTRODUÇÃO

A segurança nas escolas tem sido um tema amplamente discutido devido ao aumento frequente da violência. Nesse contexto, a tecnologia surge como uma possível solução para aprimorar esse problema, e é com base nisso que este projeto propõe o uso do reconhecimento facial como uma forma de melhorar a segurança na entrada dos alunos.

O presente trabalho aborda a questão da falta de supervisão e segurança nas escolas, o que pode resultar em danos físicos e psicológicos para alunos, professores e funcionários. A ausência de fiscalização adequada na entrada das escolas é um dos principais motivos para a vulnerabilidade desses ambientes, permitindo a entrada de pessoas não autorizadas. Esse cenário se torna ainda mais preocupante quando se considera que muitas escolas são frequentadas por crianças e adolescentes, que são mais vulneráveis a ameaças de violência e assédio.

Diante disso, é possível concluir que a validação da matrícula por meio do reconhecimento facial se mostra como uma medida eficaz para melhorar a segurança escolar, uma vez que um uniforme pode ser roubado e uma carteirinha pode ser falsificada. Além disso, a implementação desse projeto pode contribuir para aprimorar a gestão escolar, otimizando o controle de presença dos alunos e proporcionando maior segurança no ambiente escolar.

No que diz respeito aos métodos atuais de identificação de matrícula, eles se baseiam principalmente no uso de uniformes e, em algumas escolas, na utilização de carteirinhas. No entanto, esses métodos mostram-se insuficientes para solucionar as lacunas existentes na segurança escolar, uma vez que tais estratégias podem ser facilmente burladas.

Um exemplo alarmante disso foi a invasão armada de uma escola em Forquilha, no interior do Ceará, na qual falsos estudantes apareceram uniformizados, armados e ameaçaram o segurança (GCMAIS, Roberta Fontelles, 2022). Esse incidente destaca a necessidade de aprimorar a segurança interna das escolas por meio de medidas efetivas.

Diante desse contexto, o objetivo deste trabalho é explorar como melhorar a segurança interna das escolas por meio do uso de câmeras de monitoramento em um ambiente escolar. O objetivo geral é desenvolver um sistema de identificação e monitoramento utilizando a tecnologia de reconhecimento facial. Os objetivos específicos incluem a realização de pesquisas sobre os principais problemas de segurança interna nas escolas, a seleção das tecnologias mais adequadas para o desenvolvimento do aplicativo, considerando critérios de segurança e infraestrutura escolar, o projeto de uma interface de usuário que garanta usabilidade, estética e experiência satisfatória, facilitando o processo de autenticação da matrícula no aplicativo, o desenvolvimento do aplicativo com funcionalidade de reconhecimento facial, integração com o banco de dados simulado da escola e sistema de alerta automático para tentativas de acesso indevido, além da realização de testes de usabilidade e segurança com amostras de alunos reais para verificar a eficácia do sistema.

Justifica-se a realização deste estudo com base nas considerações de Marcy José (2021), que ressalta a insuficiência do uso exclusivo de uniformes como medida de segurança, uma vez que eles podem ser falsificados ou roubados. Nesse sentido, a tecnologia se apresenta como uma aliada promissora.

Considerando a falta de segurança e supervisão adequadas nas entradas das escolas, a adoção da autenticação da matrícula por meio do reconhecimento facial emerge como uma solução eficaz para esse problema. Essa abordagem permite uma validação rápida e precisa da identidade dos alunos, eliminando a necessidade de objetos passíveis de perda, falsificação ou roubo. Dessa forma, espera-se que a implementação de um sistema de reconhecimento facial contribua significativamente para aprimorar a fiscalização e a segurança nas escolas.

Quanto à metodologia adotada, foram empregadas pesquisas de campo, bibliográficas e abordagens quali-quantitativas. O objetivo dessas pesquisas é demonstrar como o projeto poderia ser implementado nas escolas, fornecendo informações que comprovem a eficácia do reconhecimento facial como método de segurança. O processo metodológico do projeto abrange seis etapas, incluindo pesquisa bibliográfica, pesquisa de campo, desenvolvimento do

sistema, testes de usabilidade e segurança, análise dos resultados e elaboração de um relatório técnico.

Em síntese, diante dos desafios relacionados à segurança nas escolas, a proposta deste trabalho consiste em utilizar o reconhecimento facial como uma ferramenta para aprimorar a segurança na entrada dos alunos. A implementação desse sistema visa mitigar os problemas de falta de supervisão e vulnerabilidade nas escolas, promovendo uma gestão mais eficiente e garantindo um ambiente escolar mais seguro para todos os envolvidos.

# 2. REFERENCIAL TEÓRICO

Este capítulo consiste no embasamento teórico, onde serão apresentados os tópicos do trabalho e as tecnologias utilizadas para o desenvolvimento da aplicação.

# 2.1 A falta de segurança nas escolas

Araújo (2023) afirma que, quando se trata de segurança nas escolas ainda há muito que precisa ser feito, para trazer paz e tranquilidade dentro do recinto escolar.

Em vista disso, é plausível afirmar que a segurança nas escolas pode se tonar mais eficaz com o uso da tecnologia, assim tornando o ambiente escolar mais seguro em relação a todos os alunos, professores e gestores escolares.

# 2.1.1 Como a tecnologia pode ajudar a segurança nas escolas

Ao longo dos anos, o avanço da tecnologia tem se intensificado e isso reflete nas escolas. Tendo em vista os diversos ataques que o ambiente escolar tem sofrido, um meio de tornar este ambiente seguro é com o auxílio da tecnologia.

O investimento em ferramentas modernas de segurança reflete nas matrículas para escolas particulares. E os gestores públicos estão se convencendo de que é preciso optar por soluções perenes, empregar os recursos públicos em soluções realmente eficazes, entendendo que adotar exclusivamente a vigilância humana é uma estratégia ultrapassada e suscetível a muitas falhas. (ULRICH, 2019).

# 2.1.2 O reconhecimento facial no ambiente escolar

Diversas escolas adotaram o meio do reconhecimento facial para o auxílio da segurança nas entradas e a otimização do ambiente escolar. Tornando assim um ambiente mais competente quando se trata de segurança. Dessa forma pode-se diminuir as chances de violências e ataques.

O foco das escolas eram as catracas eletrônicas, hoje, isso já mudou. Agora os gestores educacionais precisam se preocupar em formar barreiras de segurança que identifiquem, registrem e permitam ou não o acesso de pessoas e veículos. No momento, o reconhecimento facial é o recurso que apresenta melhores resultados para essa finalidade. (Ristow, 2019).

# 2.2 Tecnologias Utilizadas

Para o desenvolvimento da aplicação ScannerRF, foram aplicadas as seguintes tecnologias:

# 2.2.1 UML

A UML é uma linguagem visual utilizada no ramo da tecnologia, seu intuito consiste na modelagem de softwares. Utilizando-a, é possível facilitar o processo de projeto, análise e documentação de sistemas. Predominantemente utilizada por indústrias pelas suas inúmeras qualidades. A UML permite a representação visual da estrutura, comportamento e interações de uma aplicação.

Cumpre destacar que a UML não é um processo de desenvolvimento de software e tampouco está ligada a um de forma exclusiva, sendo totalmente independente, assim, ela pode ser utilizada por qualquer processo de desenvolvimento ou mesmo da forma que o engenheiro de software considerar mais adequada. (GUEDES, 2011)

# De acordo com Guedes (2011), a utilização dos diagramas precipita a descoberta de falhas, minimizando a possibilidade da descoberta de erros futuros. Os diagramas da UML têm como objetivo fornecer visões do sistema em diferentes aspectos, com propósito de projetar o software com estrutura sólida. A seguir, alguns exemplos:

# O diagrama de casos de uso é produzido durante a análise de requisitos e tem como principal objetivo apresentar uma visão geral das funcionalidades do software, além de representar as interações dos atores com o sistema.

# O diagrama de classes tem como objetivo principal representar graficamente as classes, atributos e relacionamentos em um sistema, proporcionando uma visão completa da estrutura da aplicação.

# O objetivo do diagrama de sequência é descrever a dinâmica do sistema, permitindo a visualização das interações entre os objetos desde o início até o fim das etapas da aplicação. É importante para analisar e projetar o comportamento do sistema.

# O objetivo do diagrama de atividade é representar o fluxo de ações e decisões tomadas no decorrer do processo da aplicação, possibilitando uma compreensão visual das etapas no sistema.

# 2.2.2 Astah Community

De acordo com Guedes (2011), o Astah community é uma ferramenta CASE que visa desempenhar um papel fundamental na colaboração para a execução de atividades durante o processo de engenharia de software.

O Astah é uma ferramenta de modelagem visual amplamente utilizada no mercado. Conta com recursos avançados e possui uma interface direta e intuitiva, que facilita a interação do usuário com a plataforma. Seu principal propósito é auxiliar na criação de diagramas e modelos para representar sistemas, proporcionando uma melhor comunicação e visualização dos elementos que compõem um software complexo. Nele, os usuários têm a capacidade de criar diversos tipos de diagramas, como diagramas de casos de uso, diagramas de classes, diagramas de sequência e diagramas de atividades, entre outros.

# 2.2.3 Visual Studio Code

O Visual Studio é uma das IDE mais utilizadas ultimamente, pois é um editor de fácil acesso e uso, onde diversas pessoas conseguem usar com facilidade, assim desenvolvendo suas aplicações de forma eficaz.

O Visual Studio Code é um editor de código-fonte leve, mas poderoso, que é executado na área de trabalho e está disponível para Windows, macOS e Linux. Fornece suporte interno para JavaScript, TypeScript e Node.js e com um ecossistema avançado de extensões para outras linguagens e runtimes (como C++, C#, Java, Python, PHP, Go, .NET). (MICROSOFT, 2023)

# 2.2.4 SQLITE3

Como lembra Banin (2018), o SQLite é compartilhado em diversos dispositivos distintos. Se trata de um dispositivo leve e incontestável, é muito favorável como repositório de dados para variadas aplicações.

O SQLite é um sistema de gerenciamento de banco de dados relacional de código aberto que não requer um servidor separado para funcionar. Ele armazena os arquivos do banco de dados dentro de sua própria estrutura, sendo assim adequado para websites e sistemas móveis.

# 2.2.5 Python

O Python é uma linguagem de programação desenvolvida com simples aprendizado, possui uma grande gama de desenvolvimento, podendo desenvolver aplicativos, sites, jogos, análise de dados, sistemas desktop, inteligência artificial, entre outros.

As principais características desta linguagem são:

* Multiplataforma, ou seja, existem diversas versões para os sistemas operacionais;
* Multiparadigma, é possível programar com diferentes bases como orientada a objeto, funcional ou procedural**.**

Manzano (2018) diz que, o Python traz uma possibilidade de permitir que suas práticas sejam rapidamente realizadas em seu local operacional, podendo visualizar instantaneamente a ação de uma orientação executada.

Figura 1: Exemplo de formulário de cadastro em Python

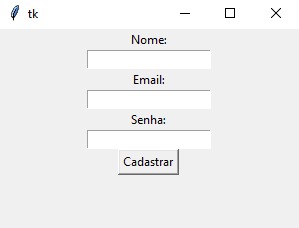


Fonte: Do próprio autor, 2023

Esse exemplo refere-se a um formulário de cadastro usando a linguagem Python, sendo esses os comandos utilizados:

* *Def*: É usado para definir uma função e é descrito como ‘*Def* \*Nome da função\*’, sua finalidade é executar a ação específica quando ela é chamada. Quando criamos um “def” criamos um objeto de função.
* *Print*: Essa função é usada para printar no terminal da IDE uma mensagem ao usuário quando não há uma tela criada ou uma aplicação para onde esta mensagem possa ser mandada.
* Botão: O comando *button* é fornecido da biblioteca TKinter e cria um botão para fazer uma determinada ação assim que ele é clicado.
* *Command*: É um comando utilizado junto com o *button*. Este comando determina a função que terá que ser executada quando o botão for clicado.
* *Pack*: É um método usado para posicionar as *widgets* (botões, caixa de texto, etc.). O pack é chamado depois da criação do *widget* e deixa o posicionamento de tal mais dinâmico e de forma automática.
* *Label*: É usado para criar um rótulo. Um rótulo exibe um texto fixo ou uma imagem, uma *label* pode ser posicionada em uma janela ou em um quadro e o texto definido dentro desse rótulo é passado pelo argumento *text*.
* *Entry*: Essa função vem da biblioteca TKinter e é usada para criar uma caixa de texto, nessa função o usuário pode inserir diversos tipos de dados de entrada.
* *Get*: Esta função obtêm o valor guardado de um objeto, como *strings* ou listas. Usando o *get* para a *string* o método retorna a própria *string* e para as listas ele retorna a declaração que foi passada.
* *MainLoop*: A função *mainloop* é usada para manter a aplicação em um loop infinito, onde a janela principal é aberta e o usuário pode fazer as interações necessárias. Essa função só retorna quando o usuário fecha o sistema.

Figura 2: Resultado do formulário de cadastro em Python

****

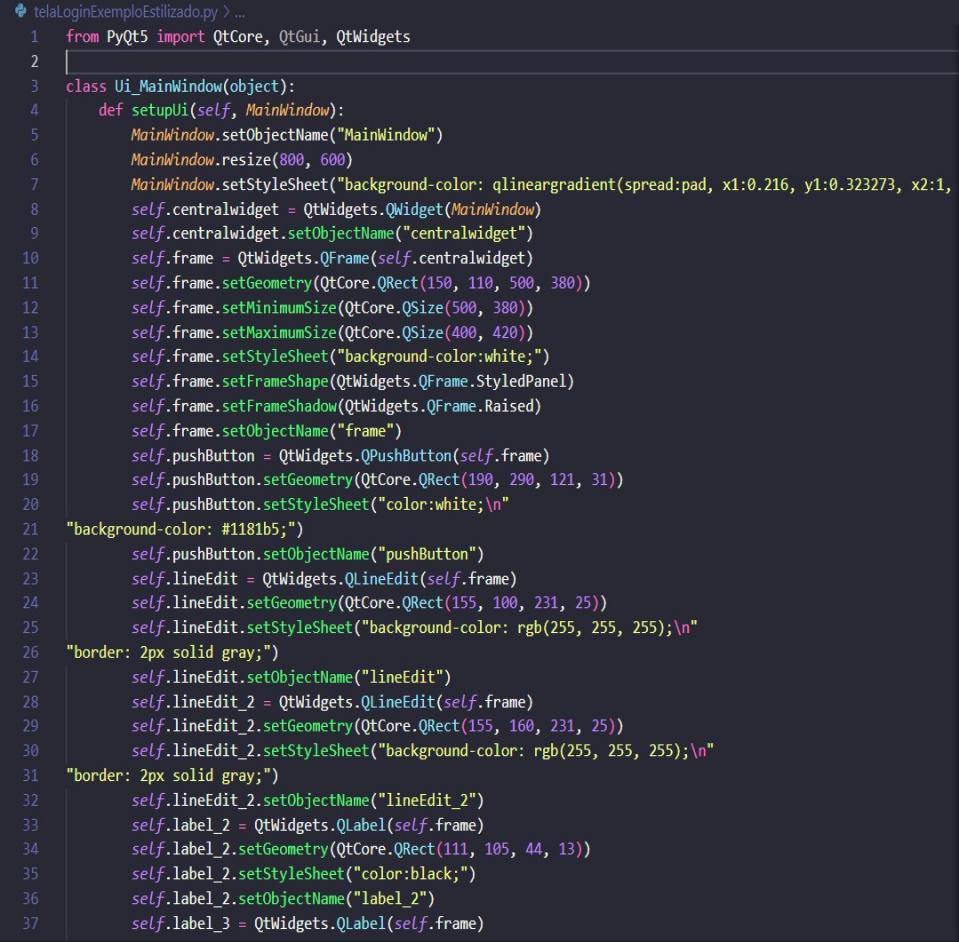
Fonte: Do próprio autor, 2023

# 2.2.6 PyQt5

Conforme Linhares (2023), o PyQt5 possibilita a conexão entre a biblioteca Qt e a linguagem Python, tornando a biblioteca PyQt5 uma interface simples para que os usuários possam programar em python.

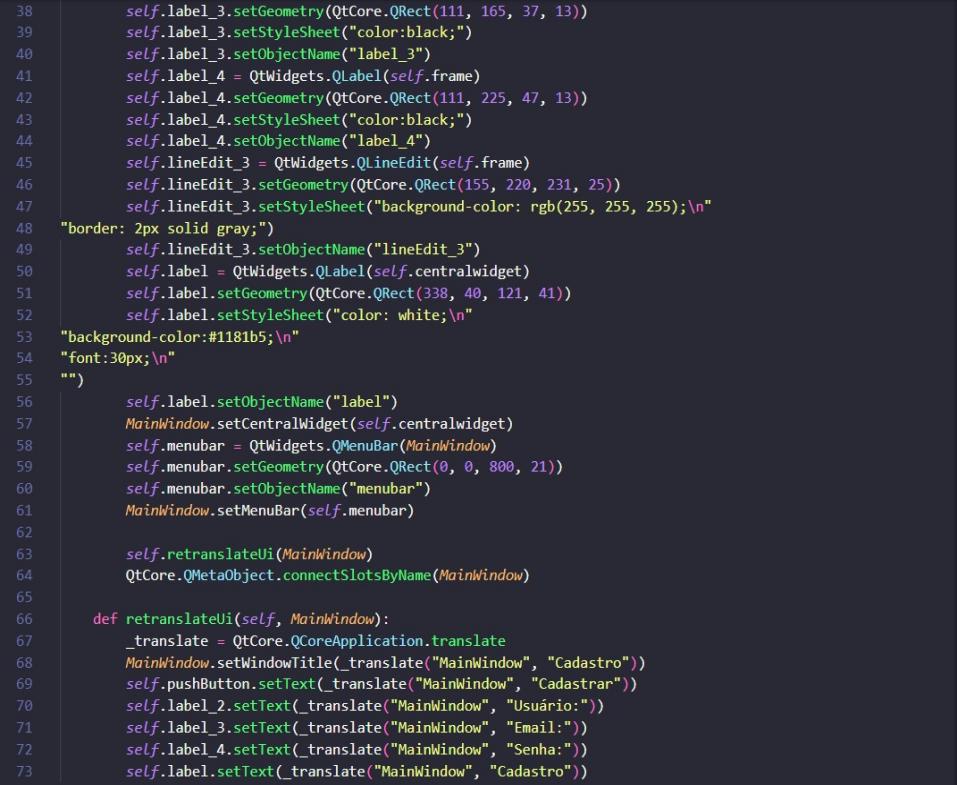
PyQt5 consiste em uma biblioteca da linguagem Python que permite o desenvolvimento aplicativos desktop. Sendo possível criar interfaces gráficas de usuário (GUIs) de forma simples e eficiente. Ele é uma ligação (*binding*) da biblioteca Qt para a linguagem Python. Esta biblioteca fornece uma ferramenta chamada Qt Designer, uma estrutura de desenvolvimento de aplicações amplamente empregada para a construção de interfaces gráficas e desenvolvimento multiplataforma de aplicativos.

Figura 3: Estilizando o código com PyQt5

****

Fonte: Do próprio autor, 2023

Figura 4: Continuação da estilização do código

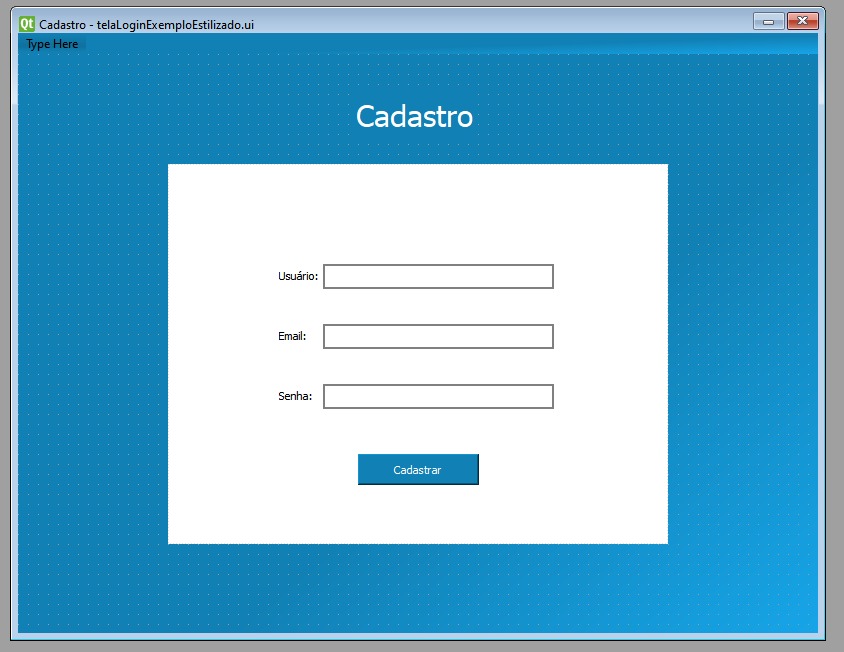


Fonte: Do próprio autor, 2023

No exemplo acima, é apresentado o código da interface do formulário. Esta aplicação foi desenvolvida’ com os seguintes comandos:

* *From*: Palavra-chave utilizada para importar módulos ou partes específicas deles em um programa.
* *MainWindow*: Um objeto que representa a janela principal de um aplicativo com interface gráfica, onde os elementos visuais são organizados.
* *Set*: Prefixo utilizado em vários métodos para configurar ou ajustar propriedades de widgets ou objetos.
* *ObjectName*: Método que atribui um nome único a um objeto na interface gráfica.
* *Resize*: Método que ajusta o tamanho de um widget para dimensões especificadas.
* *StyleSheet*: Método para aplicar estilos em *widgets*, personalizando a aparência da interface.
* *Centralwidget*: *Widget* usado como área central em janelas e interfaces gráficas.
* *Frame*: *Widget* retangular que agrupa e organiza outros widgets.
* *Geometry*: Método para definir posição e tamanho de um widget.
* *MinimumSize*: Método que define o tamanho mínimo permitido para um *widget*.
* *MaximumSize*: Método que define o tamanho máximo permitido para um *widget*.
* *FrameShape*: Método para definir a forma do contorno de um *widget*.
* *FrameShadow*: Método para definir a sombra do contorno de um *widget*.
* *Push* *Button*: Elemento de interface para criar botões clicáveis.
* *Line Edit*: Campo de entrada de texto.
* *Background-color*: Propriedade para definir a cor de fundo de um elemento.
* *Label*: Elemento para exibir texto fixo em uma interface gráfica.
* *Color*: Cor de um elemento gráfico.
* *Border*: Borda de um elemento gráfico.
* Menubar: Barra de menu que contém opções e comandos.
* *Font*: Tipografia utilizada para exibir texto.
* *WindowTitle*: Função que define o título de uma janela.
* *RetranslateUi*: Método para atualizar a tradução dos elementos da interface.
* *Translate*: Função para traduzir textos na interface.
* *Text*: Método para definir o texto exibido em um *widget*.

Figura 5: Resultado da estilização do código com PyQt5



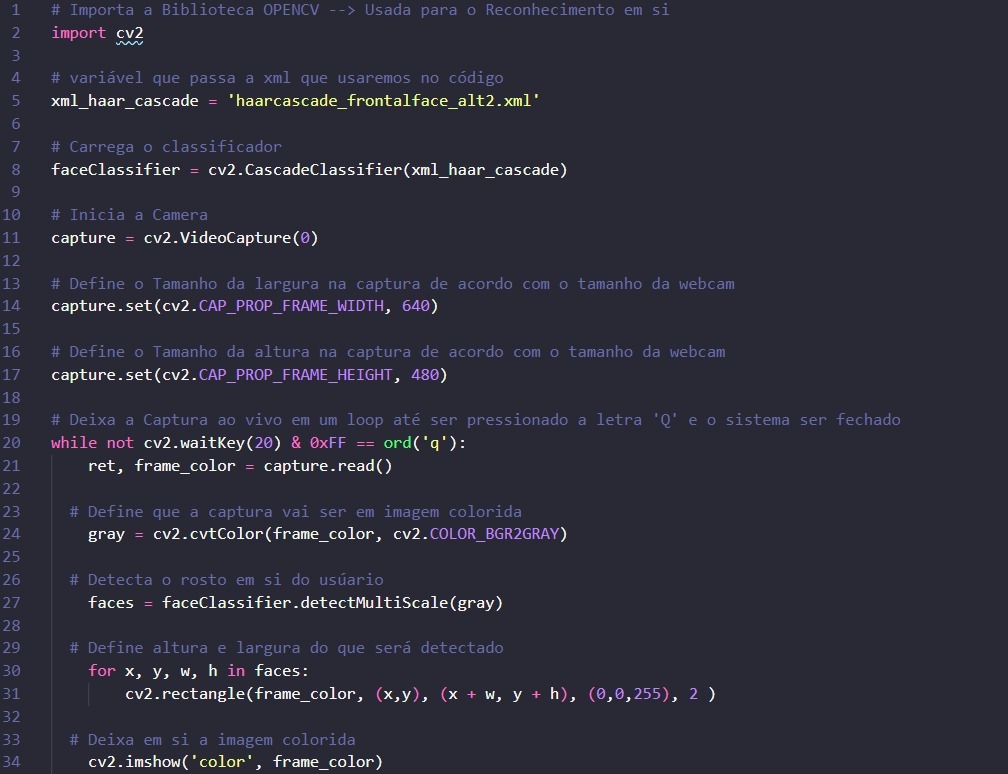
Fonte: Do próprio autor, 2023

# 2.2.7 OpenCV

OpenCV é uma biblioteca de programação que possui código aberto. Atualmente dispõe de mais de 500 funções e podem ser usadas em múltiplas linguagens como, C++, Python, Ruby, Java, entre outras.

Esta biblioteca é voltada a diversos tipos de verificação em imagens e vídeos, como o reconhecimento facial e edição de fotos e vídeos**.** A biblioteca OpenCV tem um conjunto de funções que torna simples a tarefa de manipular e processar imagens digitais. (BARELLI, 2018).

Figura 6: Exemplo de código para detecção por reconhecimento facial utilizando o OpenCV

****

Fonte: Do próprio autor, 2023

Este exemplo descreve uma detecção por meio do reconhecimento facial, utilizando os seguintes comandos:

* *Import*: Faz a importação da biblioteca OpenCV para o sistema, assim podendo usar o método da detecção facial;
* *Xml\_haar\_cascade*: Este comando trata-se da *xml* que usaremos para o sistema detectar qual o tipo de reconhecimento que ele irá fazer, se será de um objeto, rosto ou foto;
* *Capture*: inicia a câmera que está sendo utilizada, podendo ser uma webcam ou a do próprio monitor;
* *Capture.set*: Define a altura e a largura que a câmera irá detectar de acordo com o tamanho que a webcam ou a do próprio monitor alcança;
* *While* *not*: Coloca a aplicação em um loop para não fechá-la até que o usuário pressione a tecla que está na condição;
* *Ret*: É uma condição booleano que indica se a função do quadro de detecção foi bem-sucedida ou não, trazendo o *true* para bem-sucedida e *false* para malsucedida;
* *Frame\_color*: Quando o quadro de detecção é lido, essa condição o armazena em uma matriz de pixels coloridos que pode ser definido por diversas cores;
* *Cv2.cvtColor*: Está função é utilizada para realizar a conversão de cores, recebendo dois parâmetros;
* *DetectMultiScale*: Esse comando é responsável por detectar o rosto passado na câmera, podendo identificar mais de um rosto;
* *Cv2.rectangle*: Define a altura e largura do que será detectado na aplicação;
* Essa função exibe um quadro de cor na tela, recebendo dois parâmetros sendo eles o nome da janela e o *frame\_color*. O *cv2.imshow* só cria um quadro de cor na tela, porém não desfaz a execução do programa.

# 2.2.8 Figma

Trata-se de um aplicativo web para a criação de interfaces e protótipos. Sendo assim, o trabalho criado pode ser acessado por qualquer navegador, precisando somente realizar a criação da conta na plataforma.

É uma ferramenta que permite ao designer de interface a facilidade de atuar no próprio navegador, por ser um sistema alocado na nuvem e como um dos objetivos principais da ferramenta permite integração e colaboração entre a equipe que trabalhará no projeto. (MENDES, 2019)

# 3. DESENVOLVIMENTO

Nesse parágrafo abordaremos o desenvolvimento do sistema ScannerRF. Descrevendo o método UML, incluindo ilustrações de código e tela da aplicação.

# 3.1 Diagrama Entidade-Relacionamento (DER)

Segue o Diagrama de Entidade-Relacionamento (DER) do ScannerRF, tendo como objetivo representar a estrutura de banco de dados da aplicação por meio de entidades, atributos e relacionamentos.

Figura 7: Diagrama de Entidade-Relacionamento

Diagrama

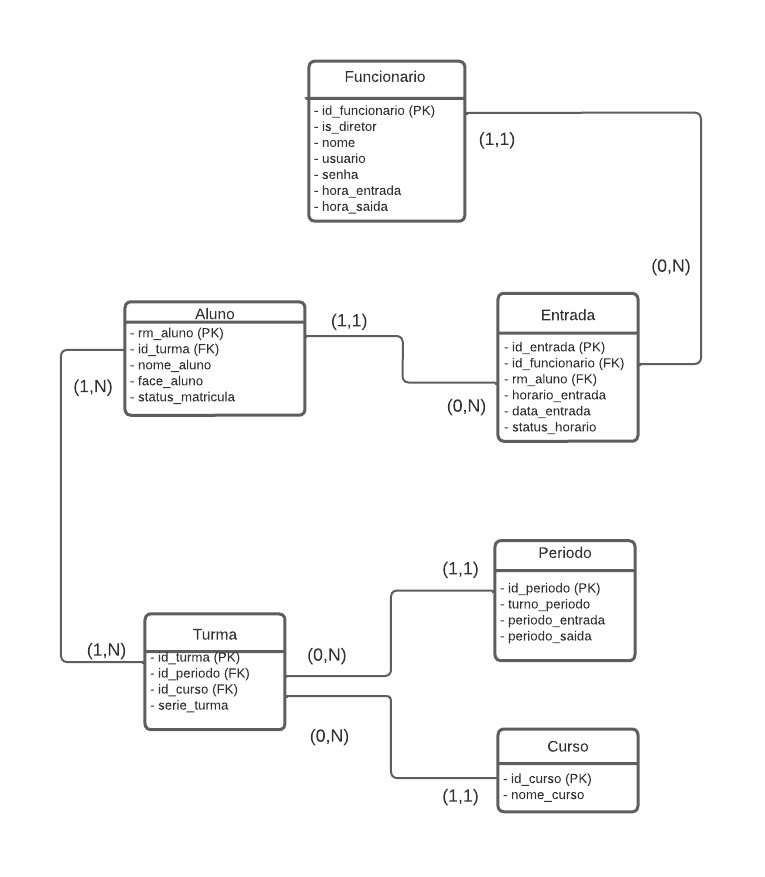
Descrição gerada automaticamente

Fonte: Do próprio autor, 2023

# 3.1.1 Modelo Entidade-Relacionamento (MER)

Segue o Modelo de Entidade e Relacionamento (MER) do ScannerRF, seu objetivo principal é representar a estrutura lógica de um sistema de banco de dados.

Figura 8: Modelo de Entidade e Relacionamento



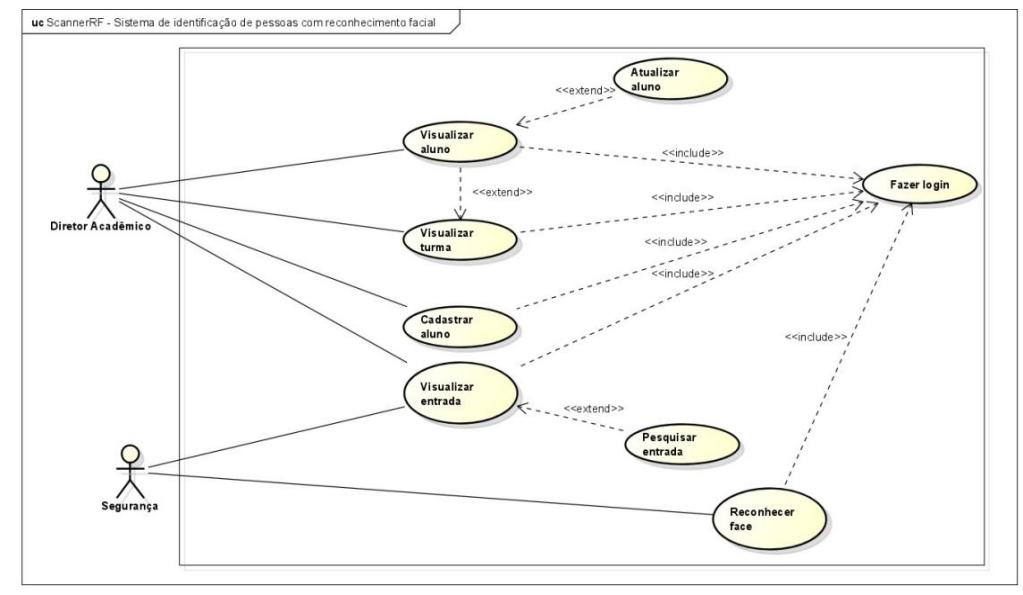
Fonte: Do próprio autor, 2023

# 

# 3.2 Diagrama de Caso de Uso

Esse diagrama foi utilizado para verificar e representar as funcionalidades ou ações do sistema. Abaixo está o caso de uso do sistema, apresentando as interações e ações que os atores poderão exercer.

Figura 9: Diagrama de Caso de Uso - ScannerRF

****

Fonte: Do próprio autor, 2023

# 3.2.1 Documentação dos Casos de Uso

Conforme orientam os módulos UML, segue a documentação do diagrama de casos de uso do ScannerRF.

Requisitos funcionais do Diretor Acadêmico:

* RF01 – O sistema deverá permitir que o diretor acadêmico efetue o login.
* RF02 – O diretor acadêmico poderá cadastrar alunos no sistema.
* RF03 – O sistema deverá permitir que o diretor acadêmico visualize as informações de todas as turmas cadastradas.
* RF04 – O diretor acadêmico poderá visualizar os alunos cadastrados no sistema.
* RF05 – O diretor acadêmico poderá atualizar as informações do aluno no sistema.
* RF06 – O sistema deverá permitir que o diretor acadêmico visualize o histórico das entradas realizadas pelos alunos, exibindo o segurança que verificou a respectiva entrada.
* RF07 – O diretor acadêmico poderá pesquisar o histórico de entradas de um aluno específico por meio do seu RM (Registro de Matrícula).

Requisitos funcionais do Segurança:

* RF01 – O sistema deverá permitir que o segurança efetue o login.
* RF02 – O sistema fornecerá ao segurança uma interface para efetuar a validação automatizada da entrada dos alunos através do reconhecimento facial, juntamente com a análise do status do horário de chegada dos estudantes.
* RF03 – O segurança poderá visualizar o histórico das entradas que ele mesmo verificou.
* RF04 – O segurança poderá buscar entradas específicas utilizando um filtro de status de horário, facilitando a identificação de entradas dentro do horário, atrasadas ou com limite excedido.

Requisitos não-funcionais da aplicação:

* RNF01 - A interface do sistema deverá ser intuitiva e de fácil utilização, garantindo uma experiência eficiente para os usuários.
* RNF02 - O reconhecimento facial deverá ser realizado em tempo real, com um tempo de resposta rápido, permitindo uma verificação ágil das entradas dos alunos.
* RNF03 - O sistema deverá suportar um grande número de alunos cadastrados e realizar consultas e atualizações de forma eficiente, garantindo um desempenho adequado mesmo em cenários de alta demanda.
* RNF04 - O sistema deverá garantir a privacidade dos dados dos alunos e adotar medidas de proteção contra acesso não autorizado.
* RNF05 - O sistema deverá ser confiável e capaz de lidar com falhas temporárias sem perdas de dados ou funcionalidades.
* RNF06 - O sistema deverá ser modular, bem documentado e de fácil manutenção.

Elaboramos quadros descritivos dos casos de uso, explicando as funcionalidades do sistema e suas interações. Essas descrições fornecem visão clara dos atores e suas ações.

Quadro 1 – Visão geral do caso de uso da aplicação a partir da perspectiva do diretor acadêmico

|  |  |
| --- | --- |
| **Nome do Caso de Uso** | **ScannerRF – Sistema de identificação de pessoas com reconhecimento facial** |
| Ator principal | Diretor acadêmico |
| Resumo | O caso de uso geral retrata a sequência de atividades executadas pelo diretor acadêmico. |
| **Cenário Principal** | |
| **Ações do Ator** | **Ações do Sistema** |
| 1. Fazer login. |  |
|  | 1. Autenticar o acesso. |
|  | 1. Apresentar a página inicial. |
| 1. Cadastrar aluno. |  |
|  | 1. Validar a correta inserção de informações nos campos. |
|  | 1. Apresentar o resultado do cadastro, indicando se foi realizado com êxito ou se ocorreu uma falha. |
| 1. Visualizar turma. |  |
|  | 1. Exibir informações da turma |
|  | 1. Listar alunos da turma. |
| 1. Visualizar aluno. |  |
|  | 1. Exibir informações do aluno. |
| 1. Atualizar aluno. |  |
|  | 1. Validar a correta inserção de informações nos campos. |
|  | 1. Apresentar o resultado da atualização, indicando se foi realizado com êxito ou se ocorreu uma falha. |
| 1. Visualizar entrada. |  |
|  | 1. Listar entradas. |
| 1. Pesquisar entrada. |  |
|  | 1. Listar histórico de entradas do aluno pesquisado. |

Fonte: Do próprio autor, 2023

Quadro 2 - Visão geral do caso de uso da aplicação a partir da perspectiva do segurança

|  |  |
| --- | --- |
| **Nome do Caso de Uso** | **ScannerRF – Sistema de identificação de pessoas com reconhecimento facial** |
| Ator Principal | Segurança |
| Resumo | O caso de uso geral retrata a sequência de atividades executadas pelo segurança. |
| **Cenário Principal** | |
| **Ações do Ator** | **Ações do Sistema** |
| 1. Fazer login. |  |
|  | 1. Autenticar o acesso. |
|  | 1. Apresentar a página inicial. |
| 1. Reconhecer face. |  |
|  | 1. Validar acesso do aluno. |
| 1. Visualizar entrada. |  |
|  | 1. Listar entradas. |
| 1. Pesquisar entrada. |  |
|  | 1. Listar histórico de entradas pelo critério pesquisado. |

Fonte: Do próprio autor, 2023

Quadro 3– Descrição do caso de uso “Fazer login” (Diretor acadêmico)

|  |  |
| --- | --- |
| **Nome do caso de uso** | **Fazer login** |
| Ator principal | Diretor acadêmico |
| Resumo | Esta descrição do caso de uso apresenta o procedimento de autenticação do diretor acadêmico no sistema. |
| **Cenário Principal** | |
| **Ações do Ator** | **Ações do Sistema** |
|  | 1. Apresentar tela do login. |
|  | 1. Requisitar credenciais de acesso (usuário e senha). |
| 1. Inserir usuário e senha nos campos fornecidos. |  |
| 1. Clicar no botão "Login". |  |
|  | 1. Verificar as credenciais inseridas. |
|  | 1. Autorizar o acesso. |

Fonte: Do próprio autor, 2023

Quadro 4 – Descrição do caso de uso “Fazer login” (Segurança)

|  |  |
| --- | --- |
| **Nome do caso de uso** | **Fazer login** |
| Ator principal | Segurança |
| Resumo | Esta descrição do caso de uso apresenta o procedimento de autenticação do segurança no sistema. |
| **Cenário Principal** | |
| **Ações do Ator** | **Ações do Sistema** |
|  | 1. Apresentar tela de login. |
|  | 1. Requisitar credenciais de acesso (usuário e senha). |
| 1. Inserir usuário e senha nos campos fornecidos. |  |
|  | 1. Verificar as credenciais inseridas. |
|  | 1. Autorizar o acesso. |

Fonte: Do próprio autor, 2023

Quadro 5– Descrição do caso de uso “Cadastrar aluno” (Diretor acadêmico)

|  |  |
| --- | --- |
| **Nome do caso de uso** | **Cadastrar aluno** |
| Ator principal | Diretor acadêmico |
| Resumo | Este caso de uso descreve o fluxo de ações para cadastrar os alunos com suas respectivas informações no sistema. |
| **Cenário Principal** | |
| **Ações do Ator** | **Ações do Sistema** |
| 1. Fazer login. |  |
|  | 1. Apresentar a página inicial. |
| 1. Selecionar a opção "Alunos" no menu superior da interface. |  |
|  | 1. Apresentar tela “Alunos”. |
| 1. Selecionar a opção “Cadastrar aluno”. |  |
|  | 1. Apresentar tela “Cadastrar aluno”. |
|  | 1. Solicitar o RM (Registro de Matrícula), o nome completo do aluno, o período, o curso, a turma do aluno e a face do aluno. |
| 1. Preencher os campos com os respectivos dados. |  |
|  | 1. Validar a correta inserção de informações nos campos. |
| 1. Confirmar cadastro. |  |
|  | 1. Apresentar o resultado do cadastro, indicando se foi realizado com êxito ou se ocorreu uma falha. |

Fonte: Do próprio autor, 2023

Quadro 6 – Descrição do caso de uso “Visualizar turma” (Diretor acadêmico)

|  |  |
| --- | --- |
| **Nome do caso de uso** | **Visualizar turma** |
| Ator principal | Diretor acadêmico |
| Resumo | Este caso de uso descreve o fluxo de ações para exibir as turmas com suas respectivas informações no sistema. |
| **Cenário Principal** | |
| **Ações do Ator** | **Ações do Sistema** |
| 1. Fazer login. |  |
|  | 1. Apresentar a interface principal. |
| 1. Selecionar a opção "Alunos" no menu superior da interface. |  |
|  | 1. Apresentar a tela “Alunos”. |
| 1. Selecionar a opção “Visualizar aluno.” |  |
|  | 1. Exibir as opções de pesquisa: "Buscar por turma" ou "Buscar por RM". |
| 1. Selecionar período desejado |  |
|  | 1. Exibir cursos do período selecionado |
| 1. Selecionar o curso desejado. |  |
|  | 1. Exibir as turmas do curso selecionado. |
| 1. Selecionar a turma. |  |
| 1. Clicar no botão “Buscar por turma” |  |
|  | 1. Apresentar a tela “Visualizar turma.” |
|  | 1. Exibir informações da turma. |
|  | 1. Listar alunos da turma. |

Fonte: Do próprio autor, 2023

Quadro 7– Descrição do caso de uso “Visualizar aluno” (Diretor acadêmico)

|  |  |
| --- | --- |
| **Nome do caso de uso** | **Visualizar aluno** |
| Ator principal | Diretor acadêmico |
| Resumo | Este caso de uso descreve o fluxo de ações para exibir os alunos com suas respectivas informações no sistema. |
| **Cenário Principal** | |
| **Ações do Ator** | **Ações do Sistema** |
| 1. Fazer login. |  |
|  | 1. Apresentar a interface principal. |
| 1. Selecionar a opção "Alunos" no menu superior da interface. |  |
|  | 1. Apresentar a tela “Alunos”. |
| 1. Selecionar a opção “Visualizar aluno.” |  |
|  | 1. Exibir as opções de pesquisa: "Buscar por turma" ou "Buscar por RM". |
| **Cenário Alternativo I - Opção “Buscar por Turma”** | |
| **Ações do Ator** | **Ações do Sistema** |
| 1. Selecionar período desejado |  |
|  | 1. Exibir cursos do período selecionado |
| 1. Selecionar o curso desejado. |  |
|  | 1. Exibir as turmas do curso selecionado. |
| 1. Selecionar a turma. |  |
| 1. Clicar no botão “Buscar por turma” |  |
|  | 1. Apresentar a tela “Visualizar turma.” |
| 1. Selecionar aluno desejado. |  |
|  | 1. Apresentar a tela “Visualizar aluno”. |
|  | 1. Exibir informações do aluno. |
| **Cenário Alternativo II - Opção “Buscar por RM”** | |
| **Ações do Ator** | **Ações do Sistema** |
| 1. Digitar RM do aluno. |  |
| 1. Clicar em “Buscar por RM”. |  |
|  | 1. Apresentar a tela “Visualizar aluno”. |
|  | 1. Exibir informações do aluno. |

Fonte: Do próprio autor, 2023

Quadro 8 – Descrição do caso de uso “Atualizar aluno” (Diretor acadêmico)

|  |  |
| --- | --- |
| **Nome do caso de uso** | **Atualizar aluno** |
| Ator principal | Diretor acadêmico |
| Resumo | Este caso de uso descreve o fluxo de ações para atualizar as respectivas informações dos alunos no sistema. |
| **Cenário Principal** | |
| **Ações do Ator** | **Ações do Sistema** |
| 1. Fazer login. |  |
|  | 1. Apresentar a tela principal. |
| 1. Visualizar aluno. |  |
|  | 1. Exibir informações do aluno. |
| 1. Selecionar o aluno desejado. |  |
| 1. Selecionar a opção “Atualizar aluno”. |  |
|  | 1. Apresentar tela “Atualizar aluno”. |
| 1. Realizar as alterações nos campos desejados. |  |
|  | 1. Validar a correta inserção de informações nos campos. |
| 1. Confirmar atualização. |  |
|  | 1. Apresentar o resultado da atualização, indicando se foi realizado com êxito ou se ocorreu uma falha. |

Fonte: Do próprio autor, 2023

Quadro 9 – Descrição do caso de uso “Visualizar entrada” (Diretor acadêmico)

|  |  |
| --- | --- |
| **Nome do caso de uso** | **Visualizar entrada** |
| Ator principal | Diretor acadêmico |
| Resumo | Este caso de uso descreve o fluxo de ações para a visualização do histórico das entradas realizadas pelos alunos. |
| **Cenário Principal** | |
| **Ações do Ator** | **Ações do Sistema** |
| 1. Fazer login. |  |
|  | 1. Apresentar a página inicial. |
| 1. Selecionar a opção “Entradas” no menu superior da interface. |  |
|  | 1. Apresentar a tela “Histórico de entradas”. |
|  | 1. Listar entradas. |

Fonte: Do próprio autor, 2023

Quadro 10 - Descrição do caso de uso “Visualizar entrada” (Segurança)

|  |  |
| --- | --- |
| **Nome do caso de uso** | **Visualizar entrada** |
| Ator principal | Segurança |
| Resumo | Este caso de uso descreve o fluxo de ações para a visualização do histórico das entradas realizadas pelos alunos. |
| **Cenário Principal** | |
| **Ações do Ator** | **Ações do Sistema** |
| 1. Fazer login. |  |
|  | 1. Apresentar a página inicial. |
| 1. Selecionar a opção “Entradas” no menu superior da interface. |  |
|  | 1. Apresentar a tela “Histórico de entradas”. |
|  | 1. Listar entradas. |

Fonte: Do próprio autor, 2023

Quadro 11 – Descrição do caso de uso “Pesquisar entrada” (Diretor acadêmico)

|  |  |
| --- | --- |
| **Nome do caso de uso** | **Pesquisar entrada** |
| Ator principal | Diretor acadêmico |
| Resumo | Este caso de uso descreve o fluxo de ações para a pesquisa do histórico de entradas de um aluno específico. |
| **Cenário Principal** | |
| **Ações** **do** **Ator** | **Ações** **do** **Sistema** |
| 1. Fazer login. |  |
|  | 1. Apresentar a página inicial. |
| 1. Selecionar a opção “Entradas” no menu superior da interface. |  |
|  | 1. Apresentar a tela “Histórico de entradas”. |
|  | 1. Listar entradas. |
| 1. Digitar o RM (Registro de Matrícula) do aluno desejado na barra de busca. |  |
|  | 1. Listar histórico de entradas do aluno pesquisado. |

Fonte: Do próprio autor, 2023

Quadro 12– Descrição do caso de uso “Pesquisar entrada” (Segurança)

|  |  |
| --- | --- |
| **Nome do caso de uso** | **Pesquisar entrada** |
| Ator principal | Segurança |
| Resumo | Este caso de uso descreve o fluxo de ações para a pesquisa do histórico de entradas de um aluno específico. |
| **Cenário Principal** | |
| **Ações** **do** **Ator** | **Ações** **do** **Sistema** |
| 1. Fazer login. |  |
|  | 1. Apresentar a interface principal. |
| 1. Selecionar a opção “Entradas” no menu superior da interface. |  |
|  | 1. Apresentar a tela “Histórico de entradas.” |
|  | 1. Listar entradas. |
| 1. Digitar o critério de status de horário (no horário, atraso ou limite excedido) desejado na barra de busca. |  |
|  | 1. Listar histórico de entradas pelo critério pesquisado. |

Fonte: Do próprio autor, 2023

Quadro 13- Descrição do caso de uso “Reconhecer face” (Segurança)

|  |  |
| --- | --- |
| **Nome do caso de uso** | **Reconhecer face** |
| Ator principal | Segurança |
| Resumo | Este caso de uso descreve o fluxo de ações para reconhecimento facial e validação de entrada do aluno. |
| **Cenário Principal** | |
| **Ações do Ator** | **Ações do Sistema** |
| 1. Fazer login. |  |
|  | 1. Apresentar a página inicial. |
| 1. Selecionar a opção “Scanner” no menu superior da interface. |  |
|  | 1. Apresentar a tela “Scanner”. |
|  | 1. Escanear face do aluno. |
|  | 1. Validar face do aluno. |
|  | 1. Verificar horário de entrada. |
|  | 1. Registrar entrada. |
|  | 1. Exibir nome, RM, período, resultado da validação de acesso e status do horário de chegada do aluno. |

Fonte: Do próprio autor, 2023

# 

# 3.3 Diagrama de Classe

Esse diagrama modela todo o comportamento da aplicação, mostrando assim tudo que ele irá utilizar e de qual forma se enquadra no sistema.

A seguir, está o diagrama de classe utilizados na aplicação do ScannerRF.

Figura 10: Diagrama de Classe - ScannerRF

Diagrama

Descrição gerada automaticamente

Fonte: Do próprio autor, 2023

**3.4 Diagrama de Sequência**

Esse diagrama foi utilizado na representação de ordem da interação dos objetos em sequências de mensagens, permitindo assim visualizar a execução da aplicação.

A seguir, estão os diagramas de sequência utilizados na aplicação do ScannerRF.

Figura 11: Diagrama de Sequência - Atualizar Aluno

Diagrama

Descrição gerada automaticamente com confiança média

Fonte: Do próprio autor, 2023

Figura 12: Diagrama de Sequência - Cadastro do Aluno

Diagrama

Descrição gerada automaticamente

Fonte: Do próprio autor, 2023

Figura 13: Diagrama de Sequência - Login

Diagrama

Descrição gerada automaticamente

Fonte: Do próprio autor, 2023

Figura 14: Diagrama de Sequência - Reconhecer Aluno

Diagrama

Descrição gerada automaticamente

Fonte: Do próprio autor,2023

Figura 15: Diagrama de Sequência do Visualizar Aluno

Diagrama

Descrição gerada automaticamente

Fonte: Do próprio autor,2023

Figura 16: Diagrama de Sequência do Visualizar Entrada

Diagrama

Descrição gerada automaticamente

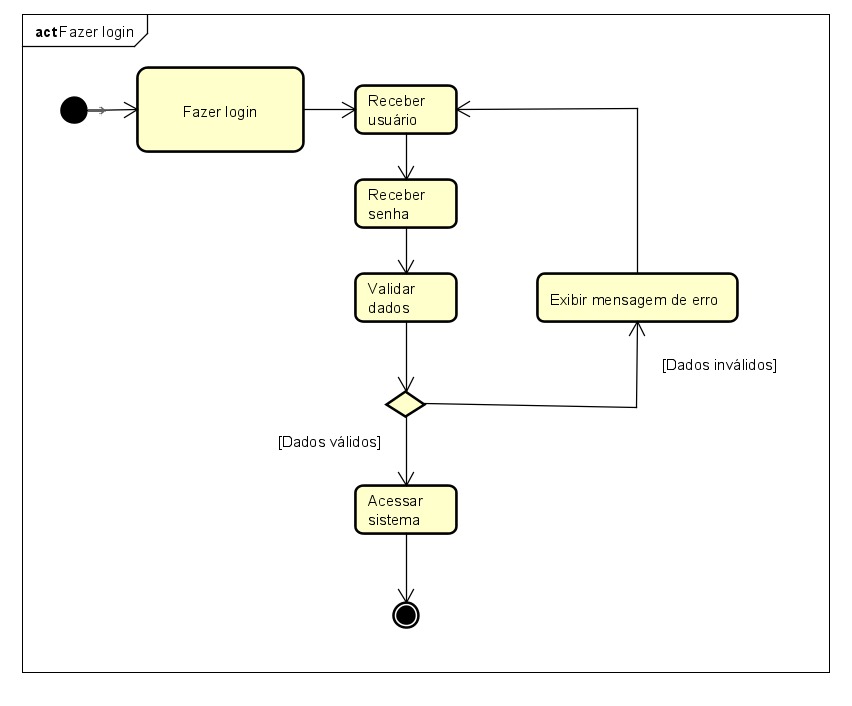
Fonte: Do próprio autor,2023

# 3.5 Diagrama de Atividade

Diagrama de Atividades:Esse diagrama foi feito para o auxílio na modelagem do fluxo de atividades dentro da aplicação, onde ele representa as ações em um desenvolvimento de negócios.

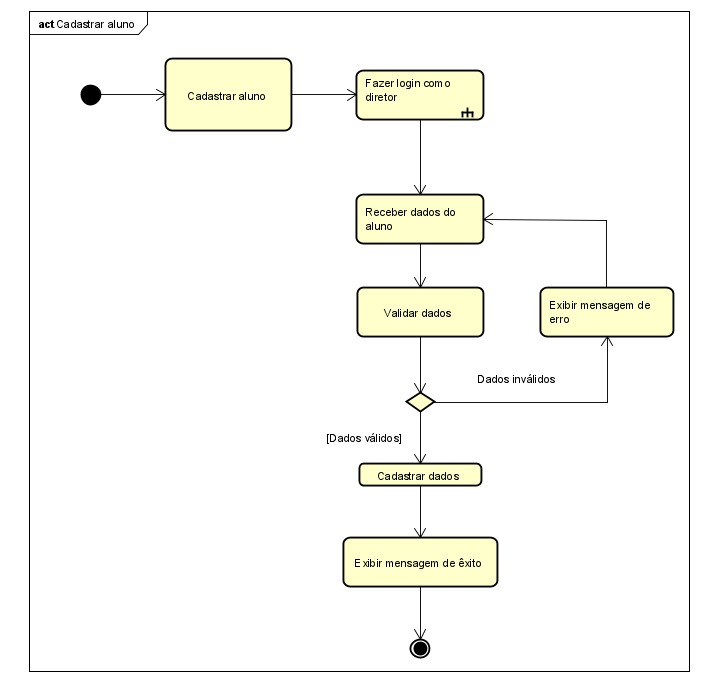
A seguir, serão apresentados os diagramas de atividades utilizados na aplicação.

Figura 17: Diagrama de Atividade Login



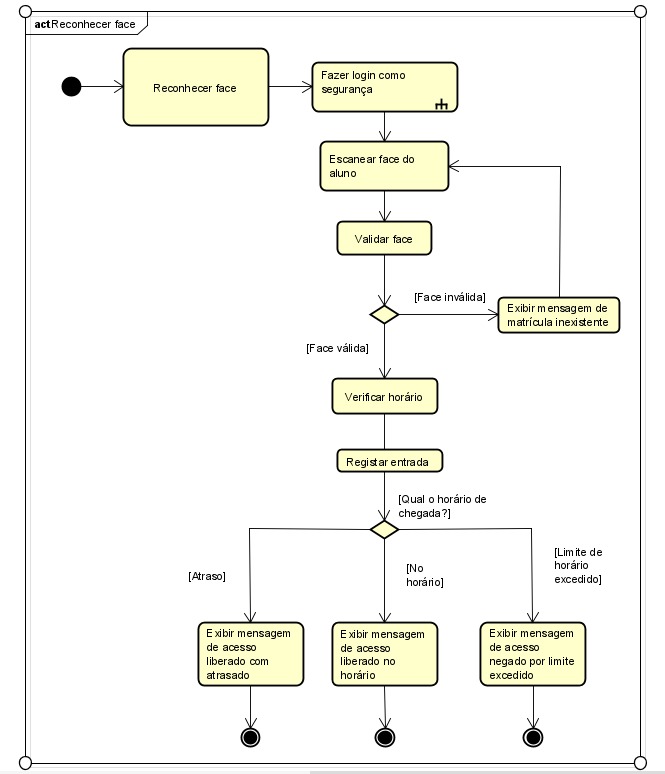
Fonte: Do próprio autor, 2023

Figura 18: Diagrama de Atividade Cadastro do Aluno



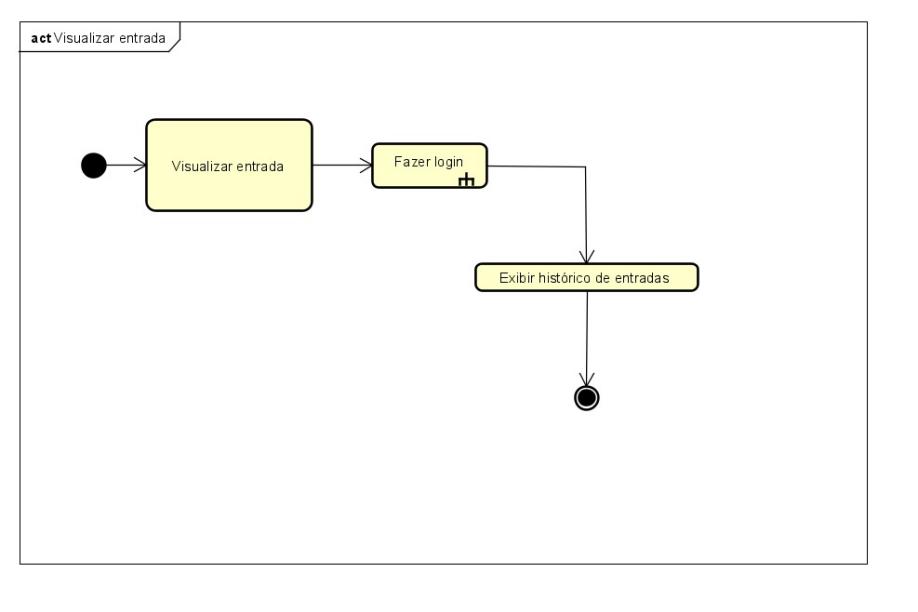
Fonte: Do próprio autor, 2023

Figura 19: Diagrama de Atividade Reconhecer Face



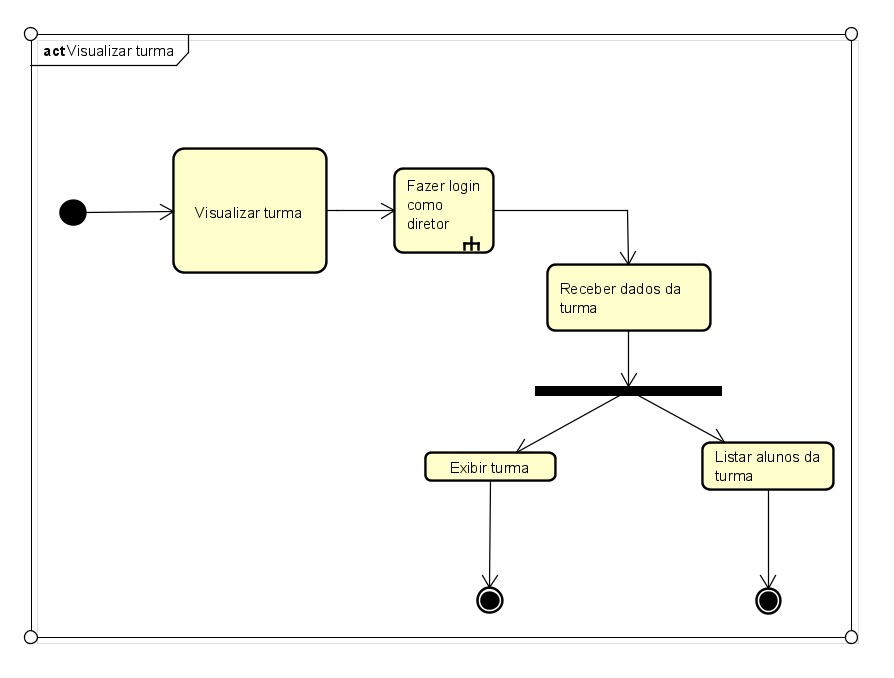
Fonte: Do próprio autor, 2023

Figura 20: Diagrama de Atividade Visualizar Entrada



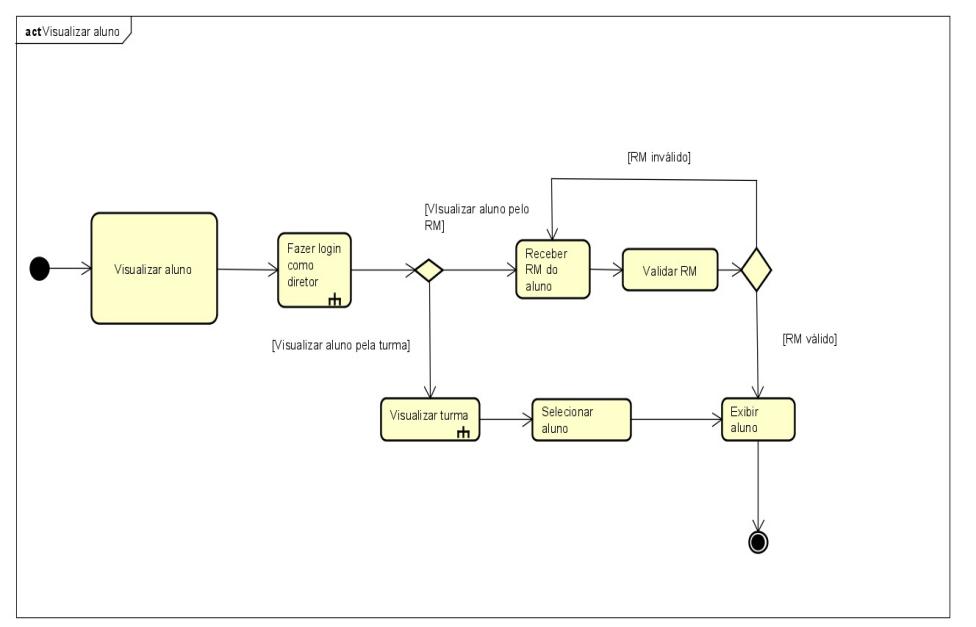
Fonte: Do próprio autor, 2023

Figura 21: Diagrama de Atividade Visualizar Turma



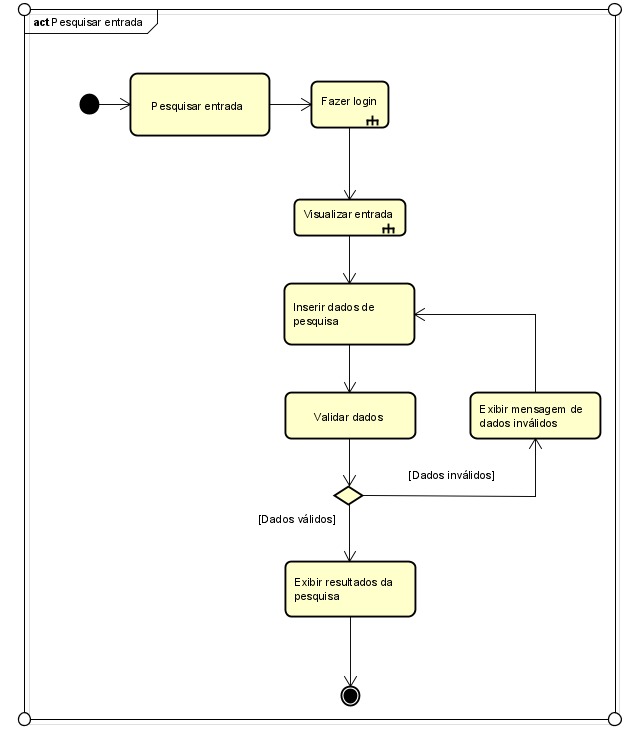
Fonte: Do próprio autor, 2023

Figura 22: Diagrama de Atividade: Visualizar Aluno



Fonte: Do próprio autor, 2023

Figura 23: Diagrama de Atividade Pesquisar Entrada



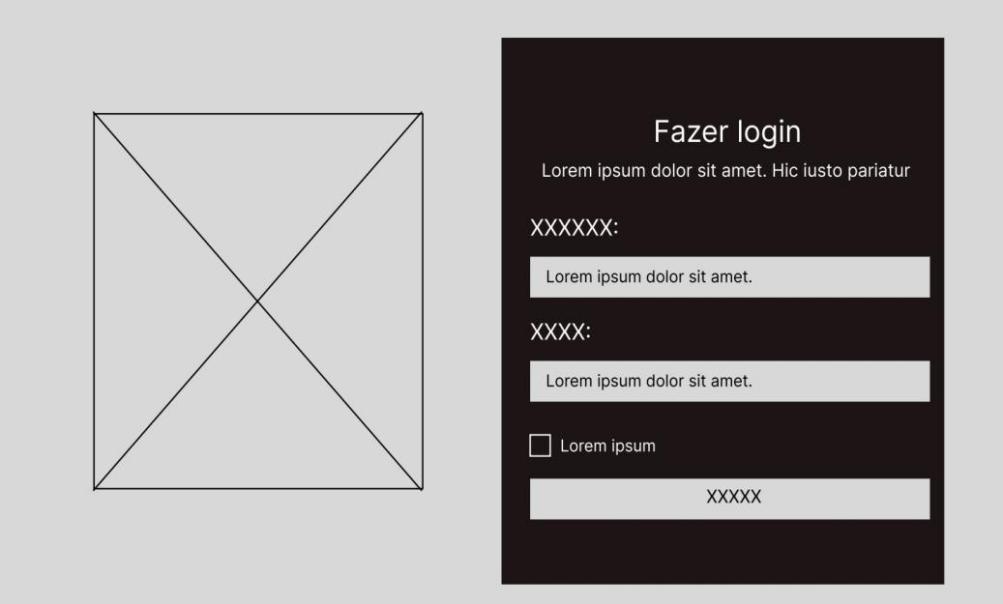
Fonte: Do próprio autor, 2023

# 3.6 Wireframe de Baixa Fidelidade

Os wireframes de baixa fidelidade são reproduções simples da aplicação, salientado no layout e na estrutura dos componentes.

A seguir será apresentado os wireframes utilizados na aplicação do ScannerRF.

Figura 24: Wireframe de Baixa Fidelidade – Login



Fonte: Do próprio autor, 2023

Esta é a interface inicial do sistema, na qual o diretor e o segurança podem efetuar o login com suas credenciais exclusivas, concedendo acesso às telas correspondentes às suas funções específicas.

Figura 25: Wireframe de Baixa Fidelidade – Página Inicial (Diretor acadêmico)



Fonte: Do próprio autor, 2023

Nesta tela, o diretor tem acesso à sua página inicial e também tem a opção de navegar para outras telas, incluindo "Alunos" e "Histórico de Entradas".

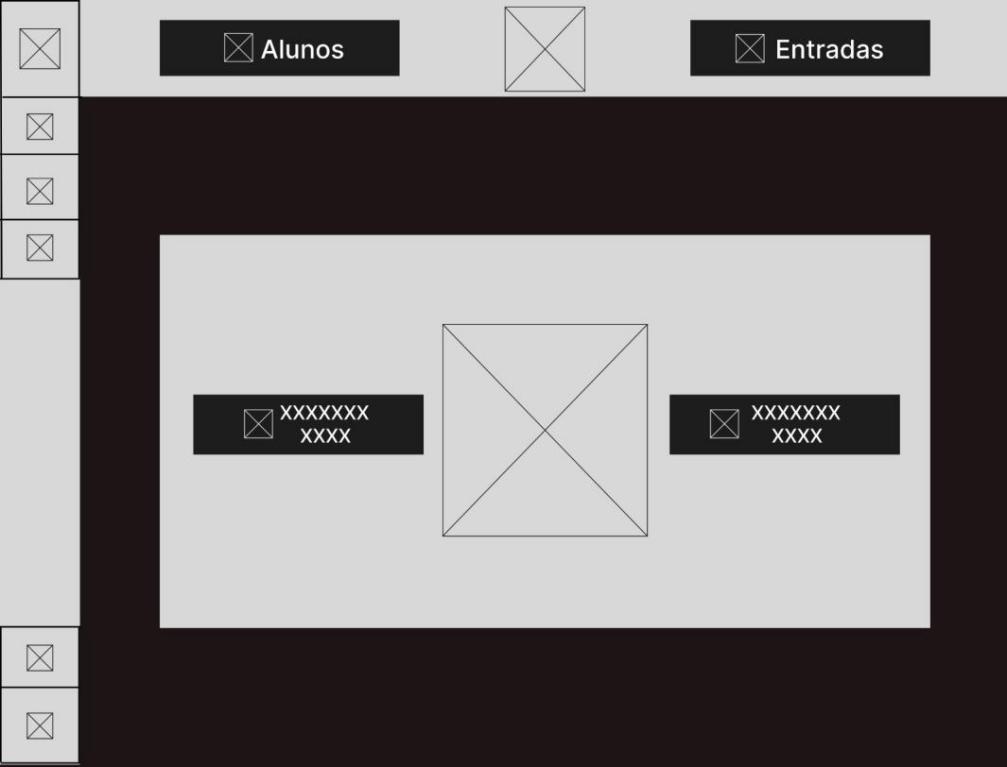
Figura 26: Wireframe de Baixa Fidelidade – Página Inicial (Segurança)



Fonte: Do próprio autor, 2023

O segurança também tem acesso a página inicial, podendo acessar somente as seguintes páginas: "Scanner", onde realizará a validação das entradas dos alunos, e "Histórico de Entradas".

Figura 27: Wireframe de Baixa Fidelidade – Alunos (Diretor acadêmico)



Fonte: Do próprio autor, 2023

O diretor pode escolher entre duas opções nesta página. Ele pode optar por acessar a página "Cadastro de Alunos" ou a página "Visualizar Alunos".

Figura 28: Wireframe de Baixa Fidelidade – Cadastrar Aluno (Diretor acadêmico)



Fonte: Do próprio autor, 2023

Nesta página, o diretor poderá efetuar o cadastro dos novos alunos matriculados.

Figura 29: Wireframe de Baixa Fidelidade – Pesquisar Aluno (Diretor acadêmico)



Fonte: Do próprio autor, 2023

O diretor possui duas opções para realizar a pesquisa. Ele pode optar por pesquisar pelo próprio RM (registro de matrícula) ou realizar uma busca mais ampla, utilizando os critérios de período, curso e turma.

Figura 30: Wireframe de Baixa Fidelidade – Visualizar Aluno (Diretor acadêmico)



Fonte: Do próprio autor, 2023

Ao realizar a busca do aluno utilizando o seu respectivo RM, o diretor terá a capacidade de visualizar as informações relativas ao aluno. Além disso, ele terá a opção de atualizar os dados do aluno, caso seja necessário.

Figura 31: Wireframe de Baixa Fidelidade – Visualizar Aluno (Diretor acadêmico)



Fonte: Do próprio autor, 2023

Ao realizar a pesquisa mais abrangente, que envolve a busca por turma utilizando os critérios de período, curso e turma, o diretor terá acesso às informações detalhadas sobre a turma. Isso inclui a quantidade de alunos, nomes dos alunos presentes na turma e informações sobre a própria turma.

Figura 32: Wireframe de Baixa Fidelidade – Atualizar Aluno (Diretor acadêmico)



Fonte: Do próprio autor, 2023

Esta página é acessada a partir da visualização individual de um aluno. O diretor tem a capacidade de alterar os dados do aluno.

Figura 33: Wireframe de Baixa Fidelidade – Histórico de Entradas (Diretor acadêmico)

Tabela

Descrição gerada automaticamente

Fonte: Do próprio autor, 2023

O diretor tem a opção de consultar o histórico de entradas dos alunos de forma geral ou pode optar por visualizar o histórico de entrada de um aluno específico, realizando a busca pelo RM (Registro de Matrícula). Essa funcionalidade permite ao diretor verificar se os alunos estão chegando atrasados ou entrando no horário correto.

Figura 34: Wireframe de Baixa Fidelidade – Histórico de Entradas (Segurança)

Tabela

Descrição gerada automaticamente

Fonte: Do próprio autor, 2023

Nesta tela, o segurança tem a capacidade de verificar se o aluno está atrasado, obtendo informações detalhadas sobre a entrada desse aluno. Isso inclui o horário em que ele entrou, o seu período, nome e seu respectivo RM.

Figura 35: Wireframe de Baixa Fidelidade – Scanner (Segurança)

Forma

Descrição gerada automaticamente

Fonte: Do próprio autor, 2023

Essa é uma funcionalidade exclusiva do segurança. Quando esta tela é aberta, ela ativa automaticamente a câmera para realizar a identificação dos alunos.

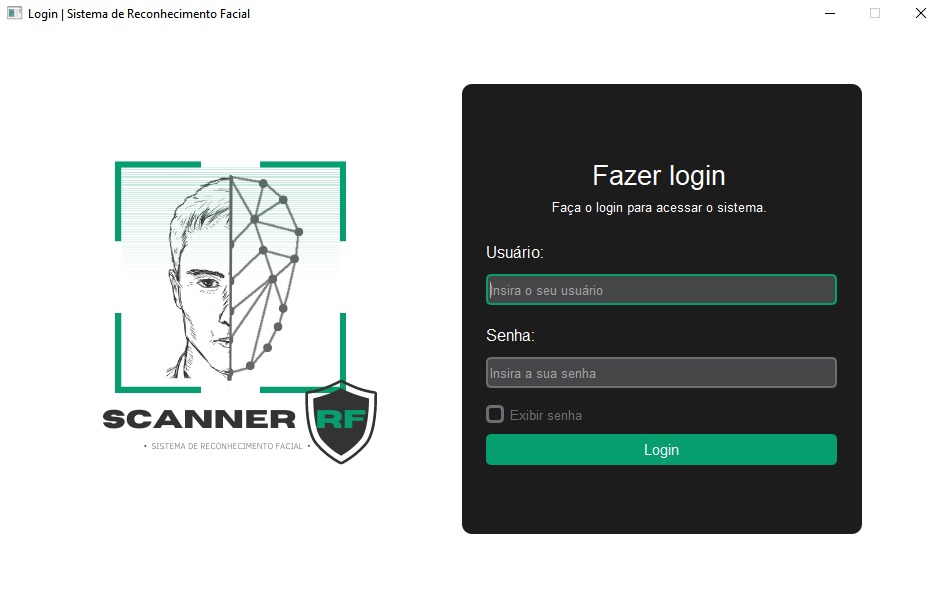
# 

# 3.7 Prototipação das Telas

A prototipação é o processo no qual um modelo de sistema é criado com a finalidade de testar e validar conceitos, funcionalidades, design, e desenvolver produtos permitindo uma validação iterativa e aperfeiçoamento constante durante o desenvolvimento de criação.

A seguir, será apresentada as telas prototipadas da aplicação ScannerRF.

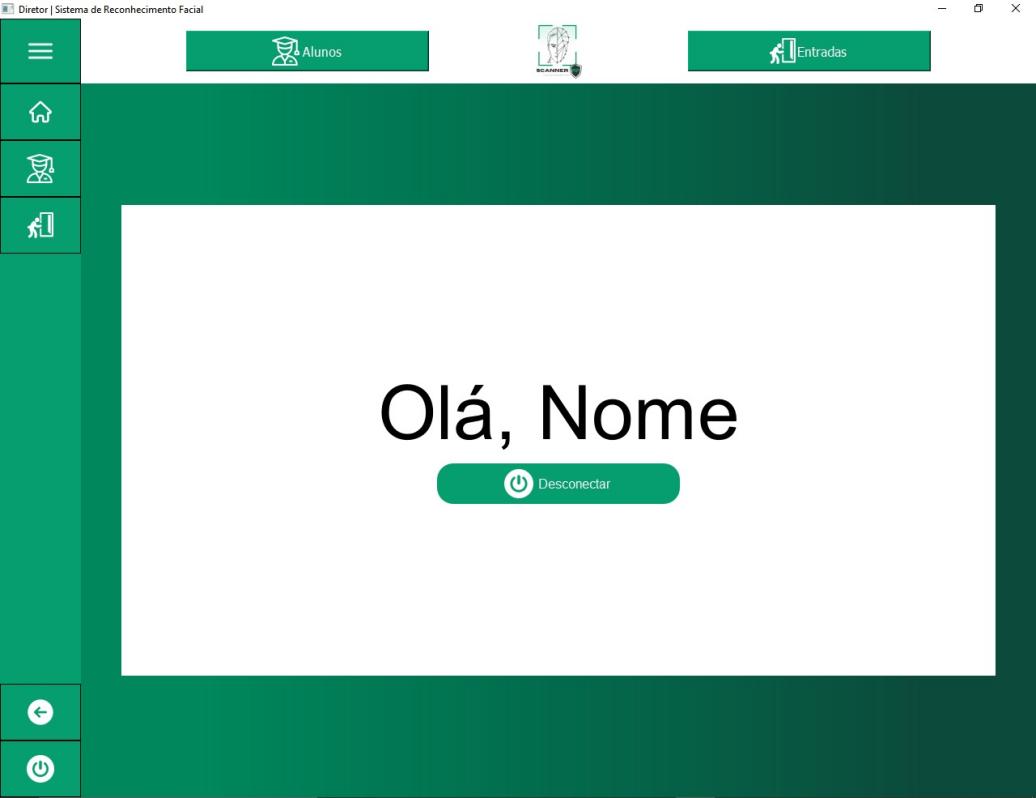
Figura 36: Prototipação - Login



Fonte: Do próprio autor, 2023

Nessa interface, é possível efetuar o login para acessar o sistema, com a presença de dois perfis de usuário: o diretor e o segurança.

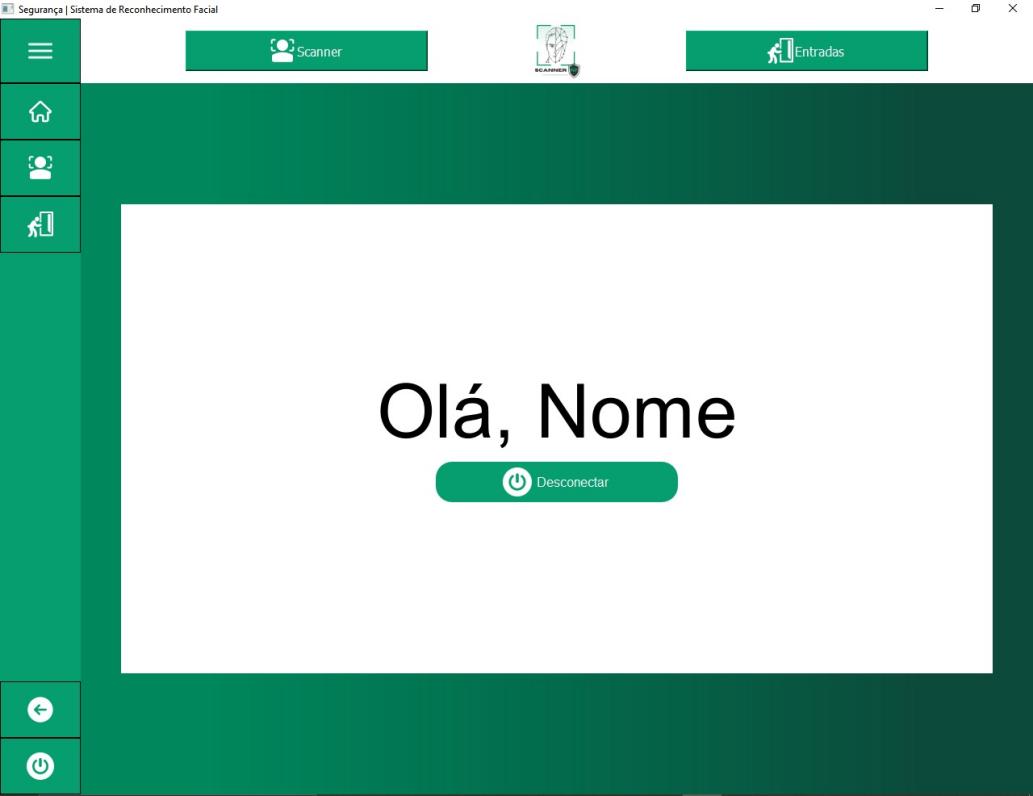
Figura 37: Prototipação - Página Inicial (Diretor acadêmico)



Fonte: Do próprio autor, 2023

Após o login ser realizado com as credenciais do diretor, essa interface será revelada como a tela inicial do sistema. Ela oferece ao usuário a opção de navegar para outras telas por meio das opções do menu, incluindo "Início", "Alunos" e "Entradas". Além disso, fornece a possibilidade de desconectar do sistema. O menu permanecerá visível em todas as páginas, proporcionando uma facilidade ao usuário para navegar pelo sistema.

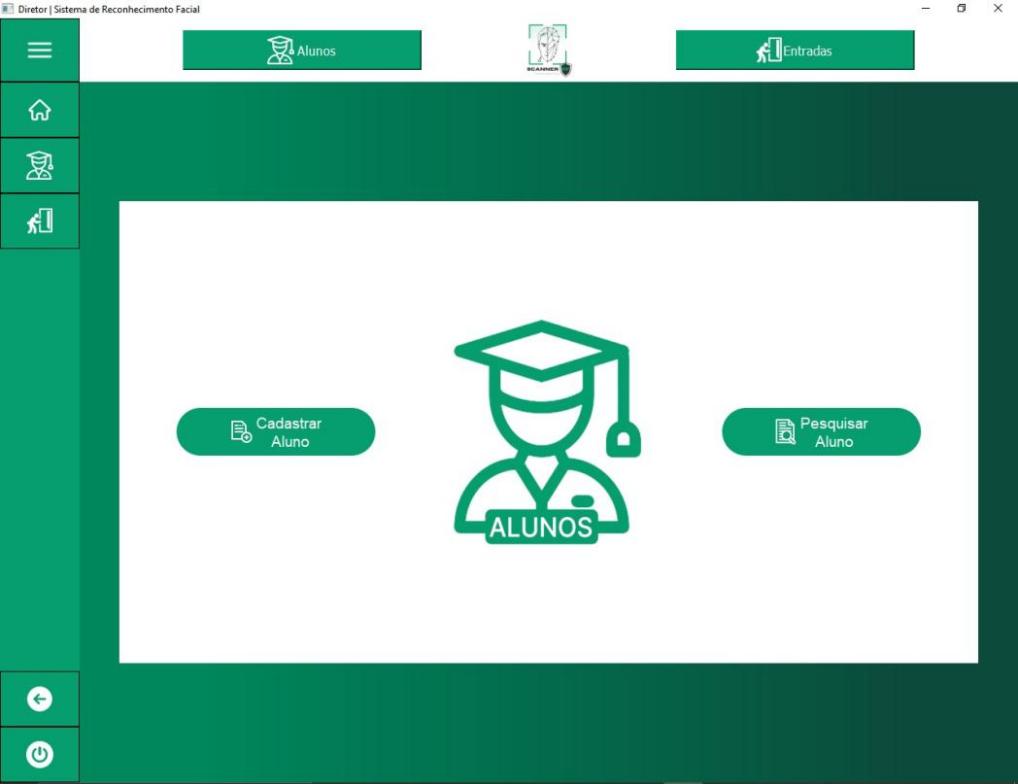
Figura 38: Prototipação - Página Inicial (Segurança)



Fonte: Do próprio autor, 2023

Essa interface será revelada ao efetuar o login com as credenciais do segurança. Além de proporcionar a capacidade de navegar para outras telas, também oferece a opção de desconectar do sistema. Devido à distinção do perfil do usuário, apresenta opções diferentes no menu, incluindo "Início", "Scanner" e "Entradas". O menu também irá permanecer visível em todas as telas.

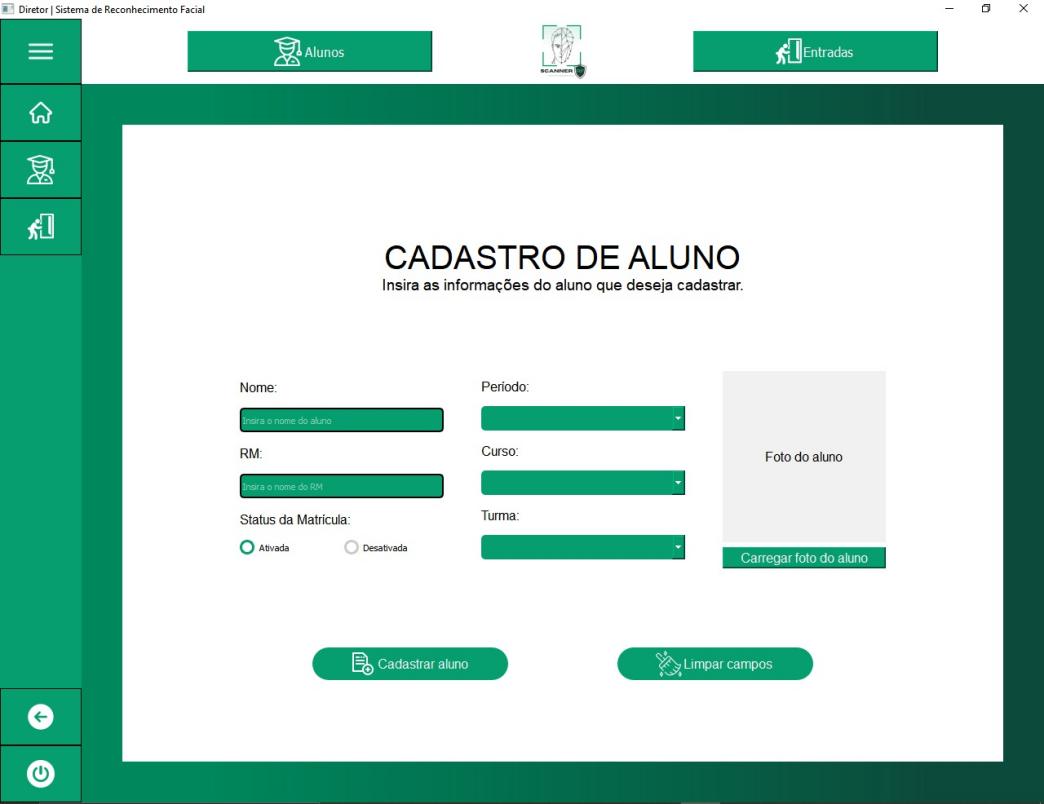
Figura 39: Prototipação - Alunos (Diretor acadêmico)



Fonte: Do próprio autor, 2023

Nessa tela, além das opções do menu, serão apresentadas mais duas opções para o usuário, possibilitando-o em optar entre acessar a página de cadastro do aluno ou realizar pesquisar aluno.

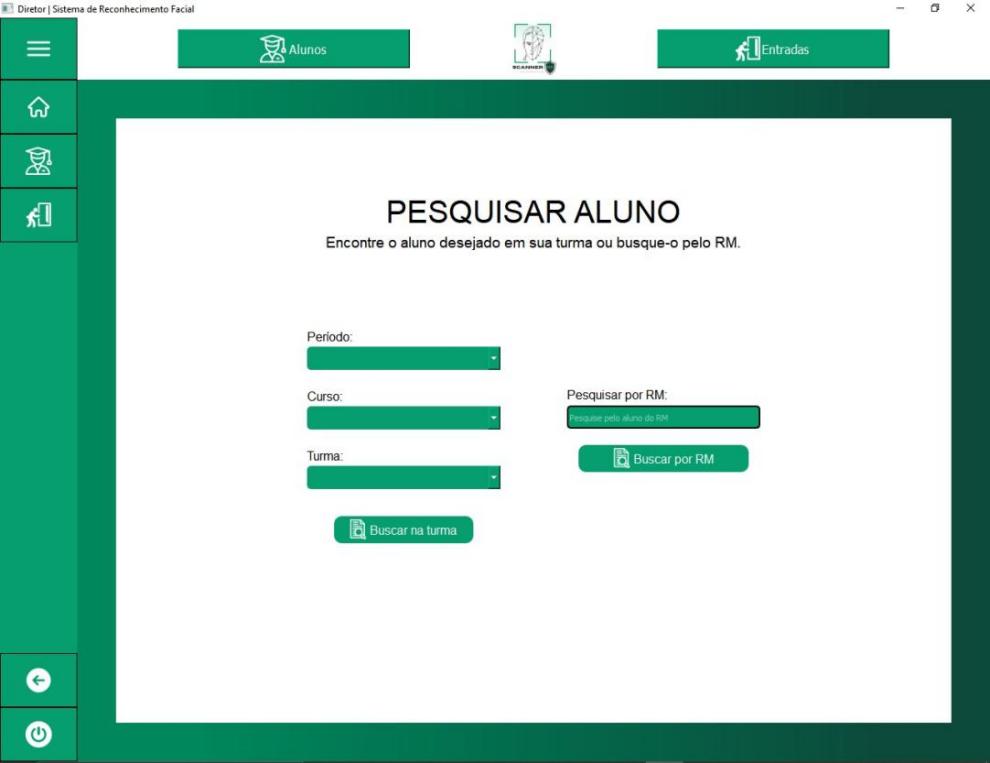
Figura 40: Prototipação - Cadastrar Aluno (Diretor acadêmico)



Fonte: Do próprio autor, 2023

Essa tela é o local destinado para realizar o cadastro dos alunos matriculados. Sendo possível preencher com os dados relacionados ao aluno, possibilitando também adicionar a foto do estudante.

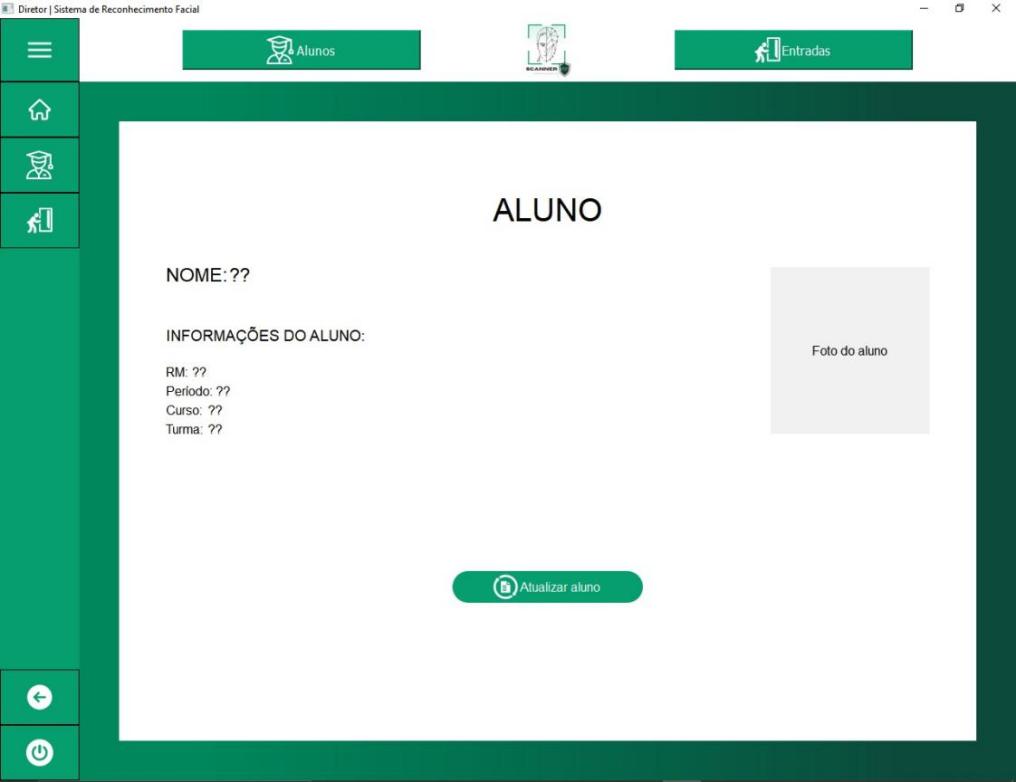
Figura 41: Prototipação - Pesquisar Aluno (Diretor acadêmico)



Fonte: Do próprio autor, 2023

Na presente tela, poderá ser realizada a pesquisa dos alunos, permitindo em optar na busca na turma, com os critérios de pesquisa "período", "curso' e "turma", ou por meio do próprio RM (registro de matrícula) do aluno.

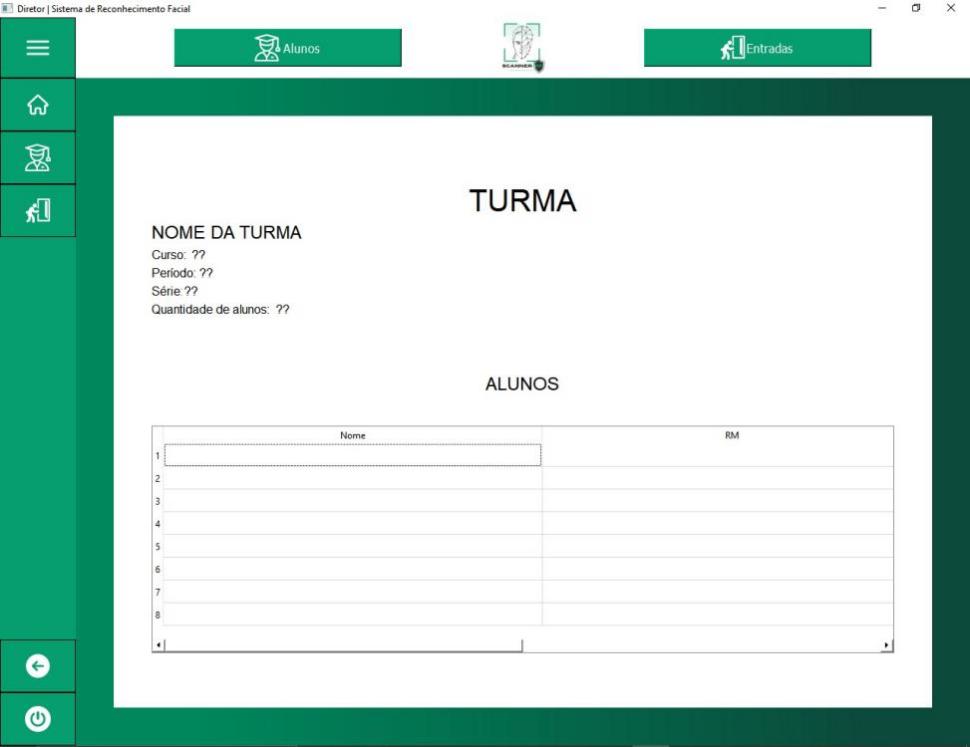
Figura 42: Prototipação - Visualizar Aluno (Diretor acadêmico)



Fonte: Do próprio autor, 2023

Ao realizar a busca utilizando o RM (Registro de matrícula) do aluno, essa tela será apresentada. Nesse contexto, é possível visualizar os dados referentes ao aluno pesquisado, além de possibilitar o usuário a se direcionar a outra tela para atualizar os dados desse aluno.

Figura 43: Prototipação - Visualizar Turma (Diretor acadêmico)

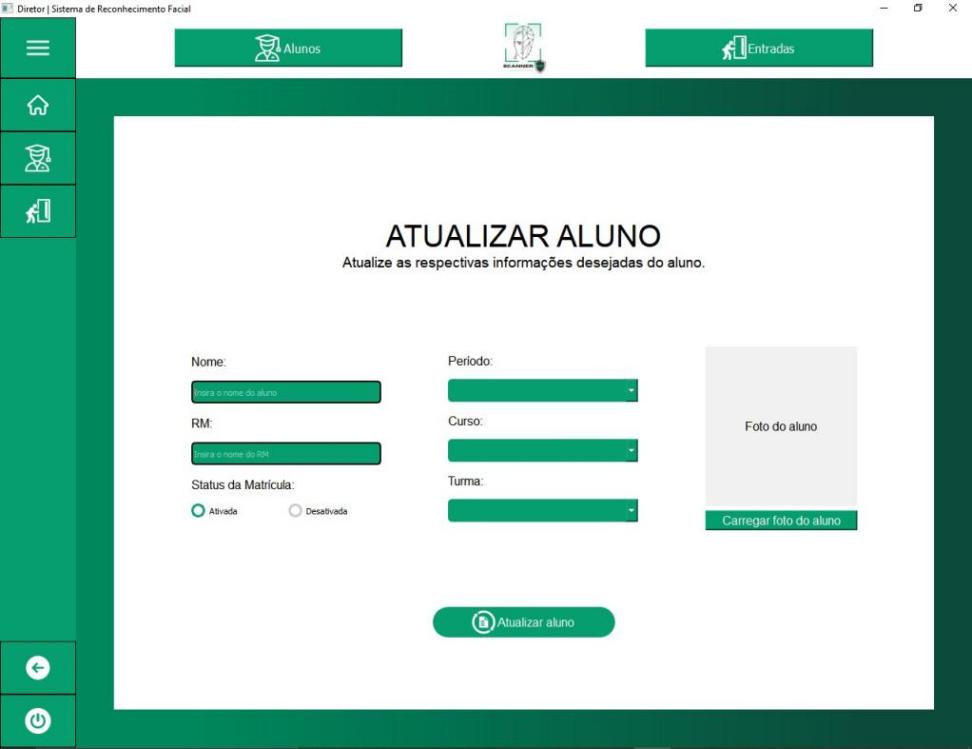


Fonte: Do próprio autor, 2023

Após realizar a busca na turma, utilizando os critérios mencionados anteriormente, essa tela irá ser apresentada. Nela irá conter algumas informações relevantes sobre a turma, acompanhada por uma tabela que lista os alunos presentes na turma pesquisada

.

Figura 44: Prototipação - Atualizar Aluno (Diretor acadêmico)



Fonte: Do próprio autor, 2023

Ao selecionar a opção "Atualizar aluno" na tela de informações do aluno, o usuário será redirecionado para essa interface específica. Onde os dados do aluno serão apresentados, oferecendo a possibilidade de realizar atualizações se necessário.

Figura 45: Prototipação - Histórico de Entradas (Diretor acadêmico)



Fonte: Do próprio autor, 2023

Nessa tela, será possível consultar o histórico de entradas dos alunos de forma geral ou individualmente, utilizando a busca por RM (registro de matrícula) do aluno.

Figura 46: Prototipação - Histórico de Entradas (Segurança)



Fonte: Do próprio autor, 2023

Nessa tela, também será possível consultar o histórico de entradas dos alunos de forma geral, possibilitando fazer a busca pelos critérios de "No horário", "Atrasado" ou "Limite excedido".

Figura 47: Prototipação – Scanner (Segurança)

Interface gráfica do usuário, Aplicativo

Descrição gerada automaticamente

Fonte: Do próprio autor, 2023

Essa é uma função do sistema utilizada somente pelo segurança. Quando essa tela é exibida, ela apresenta a câmera para realizar o reconhecimento dos alunos.

# 4. CONSIDERAÇÕES FINAIS

Ao decorrer deste trabalho, nos aprofundamos ao estudo da tecnologia de identificação por reconhecimento facial, uma solução amplamente empregada em diferentes ramos, incluindo segurança e controle de acesso. Nossa meta principal foi estudar essa tecnologia e avaliar sua eficácia para implementá-la em nosso projeto. Desenvolvemos o ScannerRF, um sistema com o propósito de simplificar a identificação de alunos e prevenir a entradas de pessoas não autorizadas no ambiente escolar.

Durante o processo, focamos na importância da gestão e controle efetivos das entradas nas escolas, proporcionando facilidade e praticidade na usabilidade do sistema, para que assim qualquer pessoa pudesse utilizá-lo. Através do ScannerRF, esperamos fornecer uma solução que facilite o trabalho dos profissionais responsáveis pela segurança e administração escolar, aprimorando a eficiência do controle do acesso.

Nossas considerações finais destacam a relevância do uso de tecnologias inovadoras, como o reconhecimento facial no ambiente escolar. Reconhecemos que, ao implementar nosso sistema, buscamos contribuir para a segurança e a ordem nas escolas, buscando um ambiente com mais proteção aos alunos e o corpo docente. Conclui-se que, nosso trabalho contribui para o avanço da segurança escolar e abre possibilidades para aprimoramentos futuros no campo da identificação biométrica em ambientes educacionais.

# 

# REFERÊNCIAS

ANDRADE, Ednalva. Ataques em Aracruz: como atirador invadiu escolas, matou quatro e fugiu: O adolescente atingiu 11 professores na Escola Estadual Primo Bitti e seguiu para uma escola particular, a cerca de um quilômetro, onde atirou em três alunos. **Ataque em escola de Aracruz**, [*s. l.*], 26 nov. 2022. Disponível em: https://www.agazeta.com.br/es/policia/ataques-em-aracruz-como-atirador-invadiu-escolas-matou-tres-e-fugiu-1122. Acesso em: 28 mar. 2023.

BANIN, Sérgio Luiz. **Python 3 Conceitos e Aplicações**: Uma Abordagem Didática. [*S. l.*: *s. n.*], 2018.

BARELLI, Felipe. **Introdução à Visão Computacional**: Uma abordagem prática com Python e OpenCV. [*S. l.*: *s. n.*], 2018. *E-book*.

FONTELLES, Roberta. Falsos estudantes invadem escola e levam colete de vigilante no Ceará: Armados, eles ameaçam o segurança. A Polícia ainda não prendeu os suspeitos. **Segurança**, [*s. l.*], 9 jun. 2022. Disponível em: https://gcmais.com.br/noticias/2022/06/09/falsos-estudantes-invadem-escola-e-levam-colete-de-vigilante-no-ceara/. Acesso em: 28 mar. 2023.

FERREIRA , Luiz Claudio. Segurança é mais do que criar barreiras à escola, dizem especialistas: Para pesquisadora, por um militar na porta não vai resolver o problema. **Educação**, [*s. l.*], 13 abr. 2023. Disponível em: https://agenciabrasil.ebc.com.br/educacao/noticia/2023-04/seguranca-e-mais-do-que-criar-barreiras-escola-dizem-especialistas. Acesso em: 12 jun. 2023.

GUEDES, Gilleanes T. A. **Uml2**: Uma Abordagem Prática. [*S. l.*: *s. n.*], 2018.

HASHTAG TREINAMENTOS. **O QUE É PYTHON E POR QUE APRENDER? GUIA PARA INICIANTES**. [*S. l.*], 31 mar. 2022. Disponível em: https://www.hashtagtreinamentos.com/o-que-e-python?gad=1&gclid=Cj0KCQjwj\_ajBhCqARIsAA37s0zXplTE\_HgOLOqWUl0ikmEPGtye7vpd6Syw7uHMgBm4mEf-F5Wvv4IaAqO7EALw\_wcB. Acesso em: 5 jun. 2023.

I3C. Tecnologia garante escolas seguras, com benefícios além de vigilância. **Segurança Eletrônica**, [*s. l.*], 2019. Disponível em: https://i3csolucoes.com.br/tecnologia-garante-escolas-seguras-com-beneficios-alem-de-vigilancia/. Acesso em: 7 jun. 2023.

LINHARES, Guilherme Trevisan Linhares. Qt e PyQt5: Qt e PyQt5. *In*: LINHARES, Guilherme Trevisan Linhares. **GUIs com Python e PyQt5 - Introdução**: GUIs com Python e PyQt5 - Introdução. Paraná: Linkedin, 22 maio 2023. Disponível em: https://www.linkedin.com/pulse/guis-com-python-e-pyqt5-introdu%C3%A7%C3%A3o-guilherme-trevisan-linhares/?originalSubdomain=pt. Acesso em: 4 ago. 2023.

MANZANO, José Augusto N. G. **Introdução à linguagem Python**. [*S. l.*: *s. n.*], 2018. Acesso em: 10 jun.2023.

MARQUES, Rogério. **OpenCV: Uma breve introdução à visão computacional com python**. [*S. l.*], 3 out. 2018. Disponível em: https://www.cedrotech.com/blog/opencv-uma-breve-introducao-visao-computacional-com-python/#. Acesso em: 13 jun. 2023.

MENDES, Fernando et al. **Projeto da Interface de Usuário da Ferramenta Notas do Ambiente Core com Usabilidade e Microinteração. Trabalho de Conclusão de Curso. -** Instituto Federal de Educação Ciência e Tecnologia de São Paulo(IFSP), Hostolândia, 2019. Disponível em: <https://hto.ifsp.edu.br/portal/images/thumbnails/images/IFSP/Cursos/Coord_ADS/Arquivos>/TCC\_Fernando\_de\_Brito\_Mendes.pdf Acesso em: 18 jul. 2023.

MICROSOFT. **Visual Studio Code**. [*S. l.*], 2023. Disponível em: https://visualstudio.microsoft.com/pt-br/. Acesso em: 19 jun. 2023.

NOGUEIRA, Antonio Sérgio. **Programando Em Python Do Básico à Web**: Acessando a Internet e Seus Serviços. [*S. l.*: *s. n.*], 2012. *E-book*.

RÁDIO E TELEVISÃO RECORD S.A. Escolas reforçam medidas de segurança após ataque em creche de Saudades (SC): Especialista indica que um controle rigoroso sobre quem entra e sai do colégio é a medida mais eficaz para evitar ataques. **Segurança nas escolas**, [*s. l.*], 5 maio 2021. Disponível em: https://recordtv.r7.com/fala-brasil/videos/escolas-reforcam-medidas-de-seguranca-apos-ataque-em-creche-de-saudades-sc-17112022. Acesso em: 28 mar. 2023.

SIMPAX. Entenda como o reconhecimento facial contribui para a segurança nas escolas. **Gestão de Pessoas**, [*s. l.*], 2019. Disponível em: https://simpax.com.br/entenda-como-o-reconhecimento-facial-contribui-para-a-seguranca-nas-escolas/. Acesso em: 14 jun. 2023.

SANTOS, Robson dos. **Atalhos VS Code: 25 Atalhos Para Aumentar Sua Produtividade**. [*S. l.*], 9 mar. 2023. Disponível em: https://www.brasilcode.com.br/atalhos-vs-code-25-atalhos-para-aumentar-sua-produtividade/. Acesso em: 12 jun. 2023.