

UNIVERSIDADE ESTÁCIO DE SÁ
CURSOS: CIÊNCIA DA COMPUTAÇÃO / ANÁLISE E DESENVOLVIMENTO
DE SISTEMAS
UNIDADE NOVA AMÉRICA

TRABALHO DE PYTHON
(DEV. RÁP. DE APLICAÇÕES EM PYTHON)

Rio de Janeiro – Rj

06/2025

Daniel Vinicius Lima Caravelos (202302249612)

Erick Freire dos Santos (202302249647)

Igor Pace da Motta (202303533292)

TRABALHO DE PYTHON

(DEV. RÁP. DE APLICAÇÕES EM PYTHON)

Trabalho de RAD apresentado a
Universidade Estácio de Sá, como
exigência para avaliação na disciplina
Desenvolvimento rápido de aplicações em
python

Orientador: Prof. Ronaldo Candido dos
Santos

SUMÁRIO

1.	INTRODUÇÃO	3
1.1	DESCRIÇÃO DO PROBLEMA	3
1.2	OBJETIVOS	4
2.	DESENVOLVIMENTO	4
2.1.	REQUISITOS DO SISTEMA	4
2.1.1.	FUNCIONAIS	4
2.1.2.	NÃO FUNCIONAIS	5
2.2.	MÓDULOS PYTHON	5
2.3.	DIAGRAMA DE CASO DE USO	5
2.4.	DIAGRAMA DE CLASSE	6
2.5.	MODELAGEM DE DADOS	6
2.5.1.	CONCEITUAL	6
2.5.2.	LÓGICO	7
2.5.3.	Sistema em funcionamento.....	7
	Sistema em funcionamento	8
2.6.	ITENS	9
2.7.	ATUALIZAÇÃO DAS VERSÕES	10
3.	CONCLUSÃO	11
	REFERÊNCIAS	12

1. INTRODUÇÃO

É com entusiasmo que apresentamos nosso trabalho, que surge da necessidade crescente de otimização e controle dos fluxos de entrada e saída de usuários em ambientes corporativos. Neste contexto, desenvolvemos um programa em Python utilizando a abordagem de Desenvolvimento Rápido de Aplicações (RAD), visando facilitar o registro e monitoramento desses fluxos, fundamentais para a segurança e gestão eficiente de espaços compartilhados.

A escolha deste tema foi motivada pela constatação de uma lacuna significativa nas ferramentas disponíveis para realizar esse tipo de controle. Muitas organizações enfrentam desafios na gestão de acesso de pessoas a diferentes áreas internas, e a falta de soluções eficazes pode resultar em dificuldades operacionais, falhas de segurança e perda de produtividade.

1.1 DESCRIÇÃO DO PROBLEMA

Uma das principais situações que levaram à realização deste trabalho é a ausência de ferramentas adequadas para registrar e monitorar o movimento de pessoas entre diferentes setores ou áreas dentro de uma empresa. Muitas vezes, os métodos tradicionais de controle, como o uso de registros em papel ou sistemas manuais, mostram-se obsoletos, imprecisos e suscetíveis a erros humanos.

Diante dessas situações e desafios, surge a necessidade premente de um programa que utilize abordagens modernas de desenvolvimento, como o Desenvolvimento Rápido de Aplicações (RAD), para oferecer uma solução ágil, confiável e personalizável para o registro de entrada e saída de usuários.

A gestão eficiente dos fluxos de entrada e saída de usuários em ambientes corporativos é uma demanda crucial para garantir a segurança, a organização e a produtividade dos espaços compartilhados.

1.2 OBJETIVOS

O principal objetivo deste trabalho foi desenvolver um programa em Python utilizando a abordagem de Desenvolvimento Rápido de Aplicações (RAD), que seja capaz de registrar de forma precisa e automatizada a entrada e saída de usuários de um setor para o outro em ambientes corporativos. Para alcançar esse objetivo, delineamos os seguintes resultados a serem obtidos:

1. Desenvolvemos um programa funcional que permita o registro eficiente das entradas e saídas de usuários, capturando informações como a data do dia e o horário de cada evento.
2. Implementamos uma interface intuitiva e amigável que facilite a utilização do programa por parte dos usuários, garantindo uma experiência de uso fluida e eficaz.
3. Integramos funcionalidades de personalização que permitam adaptar o programa às necessidades específicas de cada organização, incluindo a configuração de políticas de acesso, horários de funcionamento e estrutura de espaços físicos.
4. Garantimos a segurança e a confiabilidade dos dados registrados, adotando medidas adequadas para proteger as informações sensíveis e prevenir possíveis falhas ou violações de segurança.
5. Avaliamos a eficácia e a usabilidade do programa por meio de testes práticos e feedback dos usuários, buscando identificar possíveis melhorias e refinamentos para futuras versões.

Ao atingir esses resultados, esperamos fornecer uma solução eficiente e robusta para a gestão de acessos em ambientes corporativos, contribuindo para a melhoria da segurança, organização e produtividade desses espaços.

2. DESENVOLVIMENTO

2.1. REQUISITOS DO SISTEMA

2.1.1. FUNCIONAIS

O sistema deve registrar a entrada e a saída dos funcionários, incluindo

seu nome, setor e os horários que houve cada registro;

O sistema deve ter níveis de acesso, separando o usuário limitado do administrador, que pode incluir, excluir, alterar e consultar os funcionários e seus usuários, locais e setores;

2.1.2. NÃO FUNCIONAIS

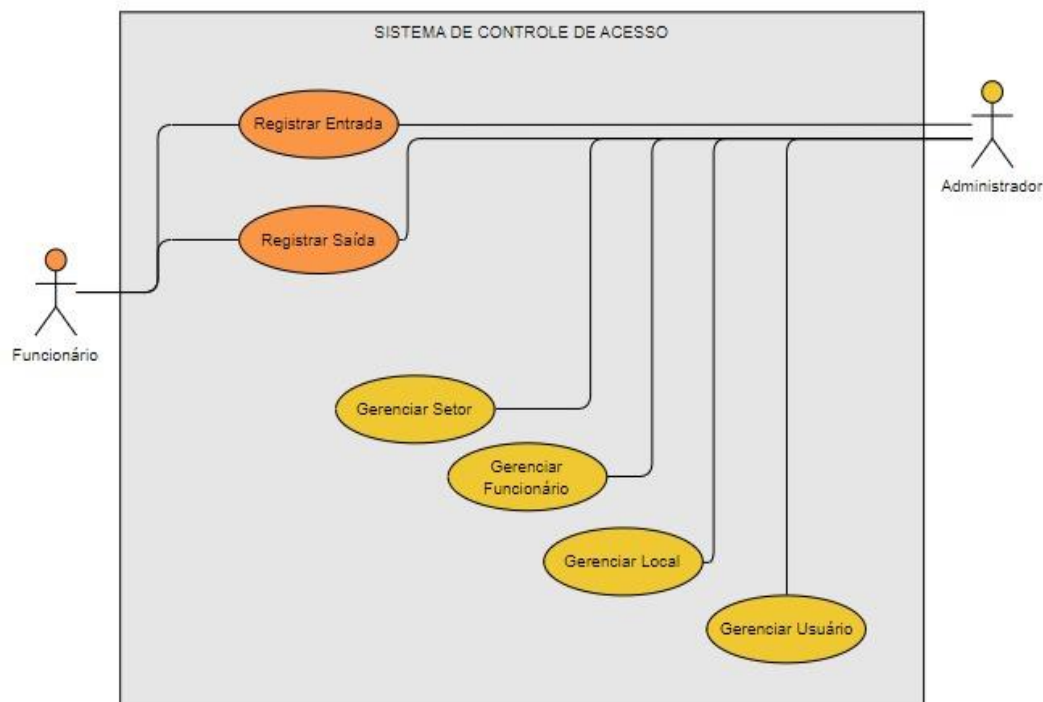
O sistema deve ser multiplataforma;

O sistema deve funcionar offline;

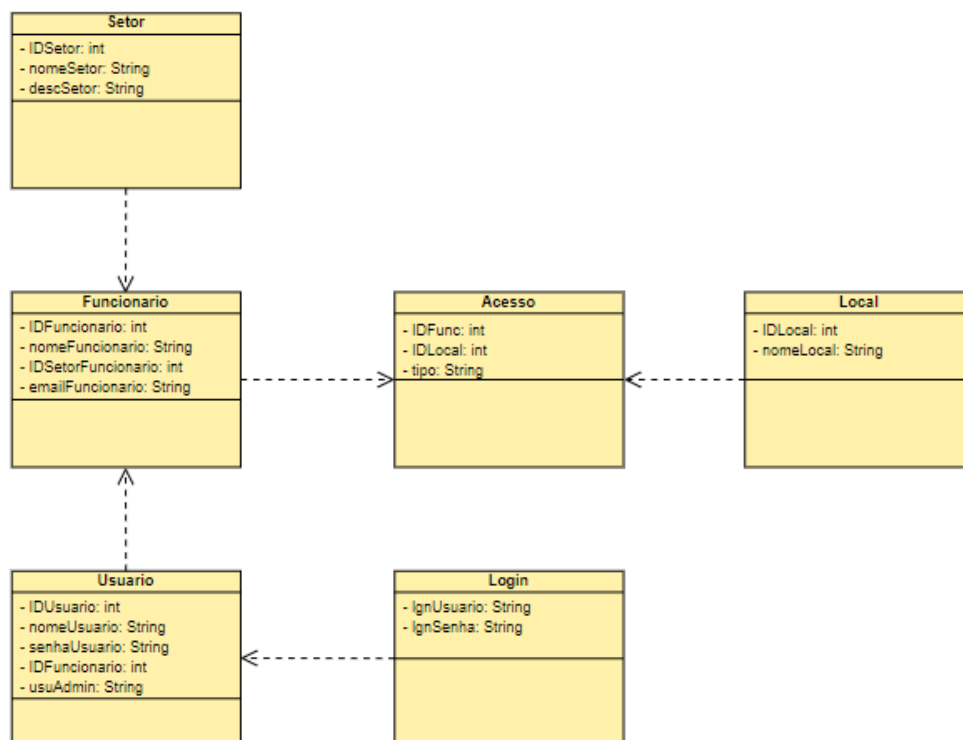
2.2. MÓDULOS PYTHON

- customtkinter
- CtkMessageBox
- PIL

2.3. DIAGRAMA DE CASO DE USO

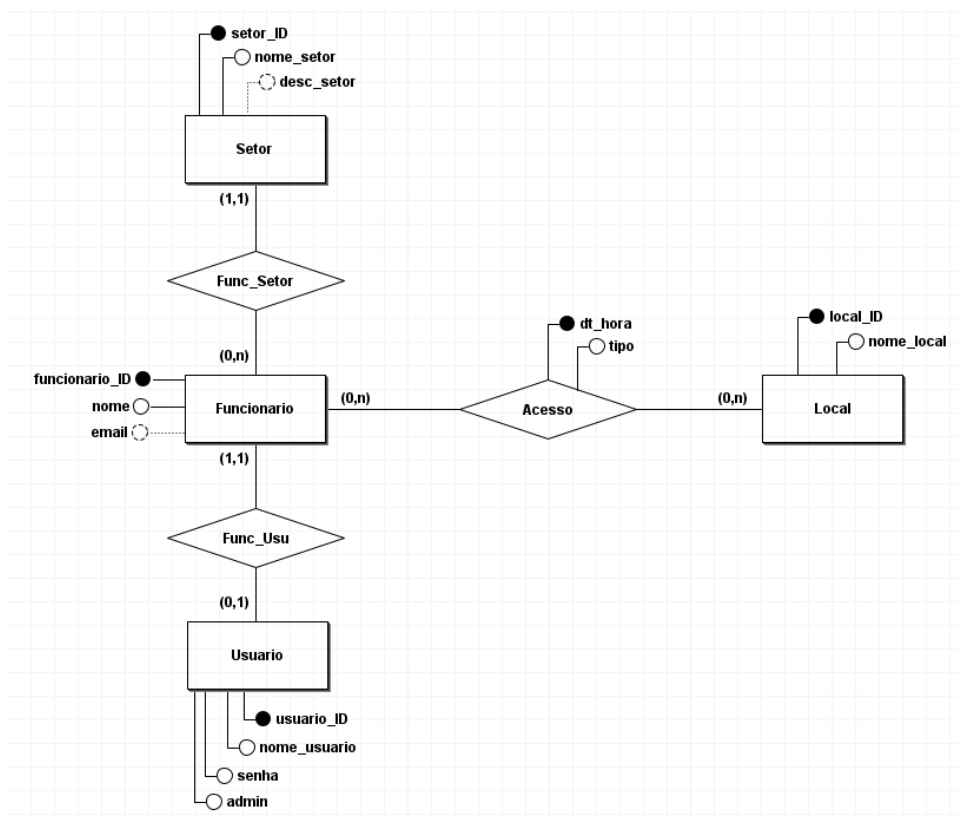


2.4. DIAGRAMA DE CLASSE

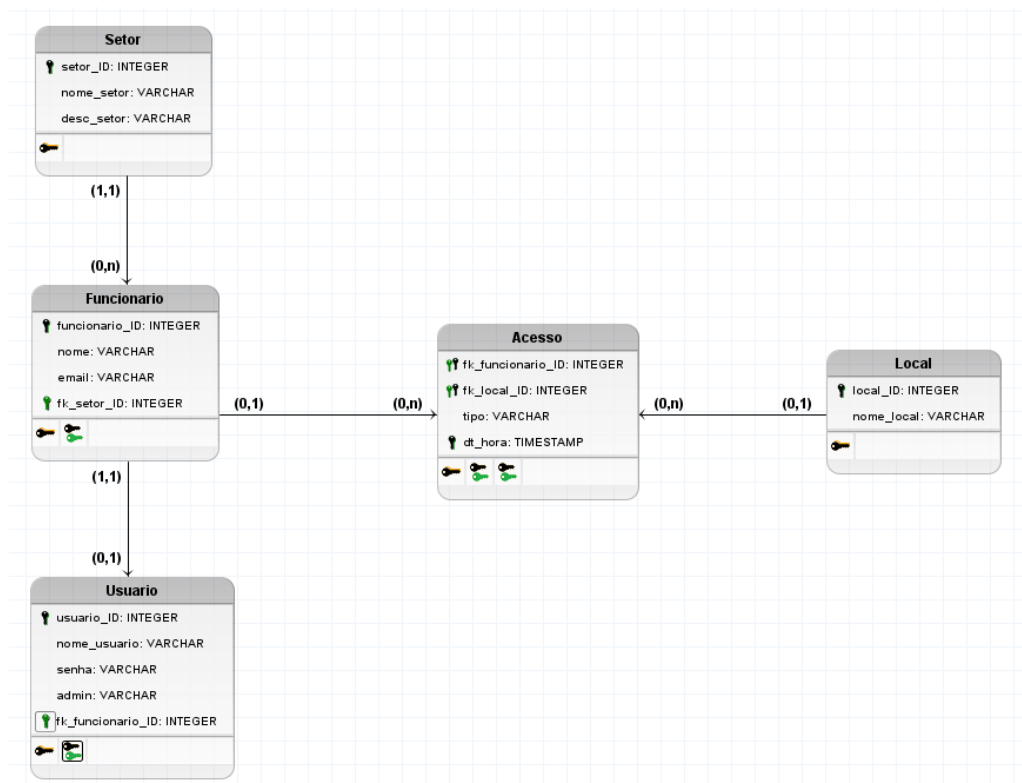


2.5. MODELAGEM DE DADOS

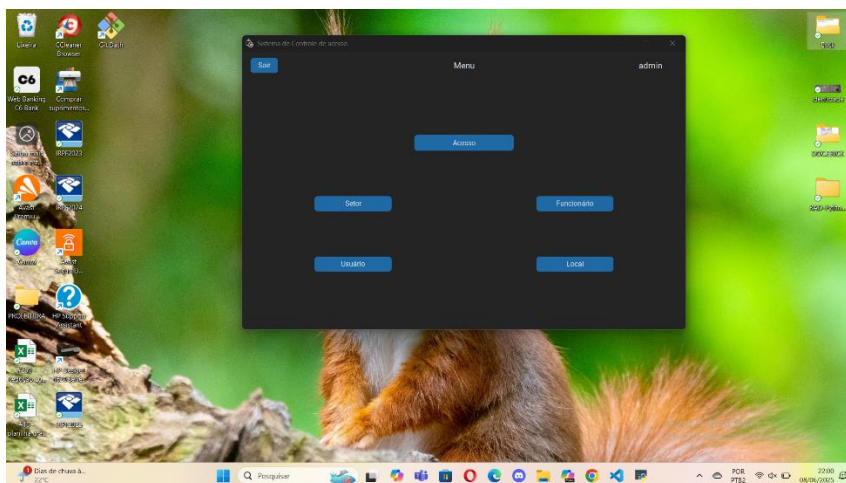
2.5.1. CONCEITUAL

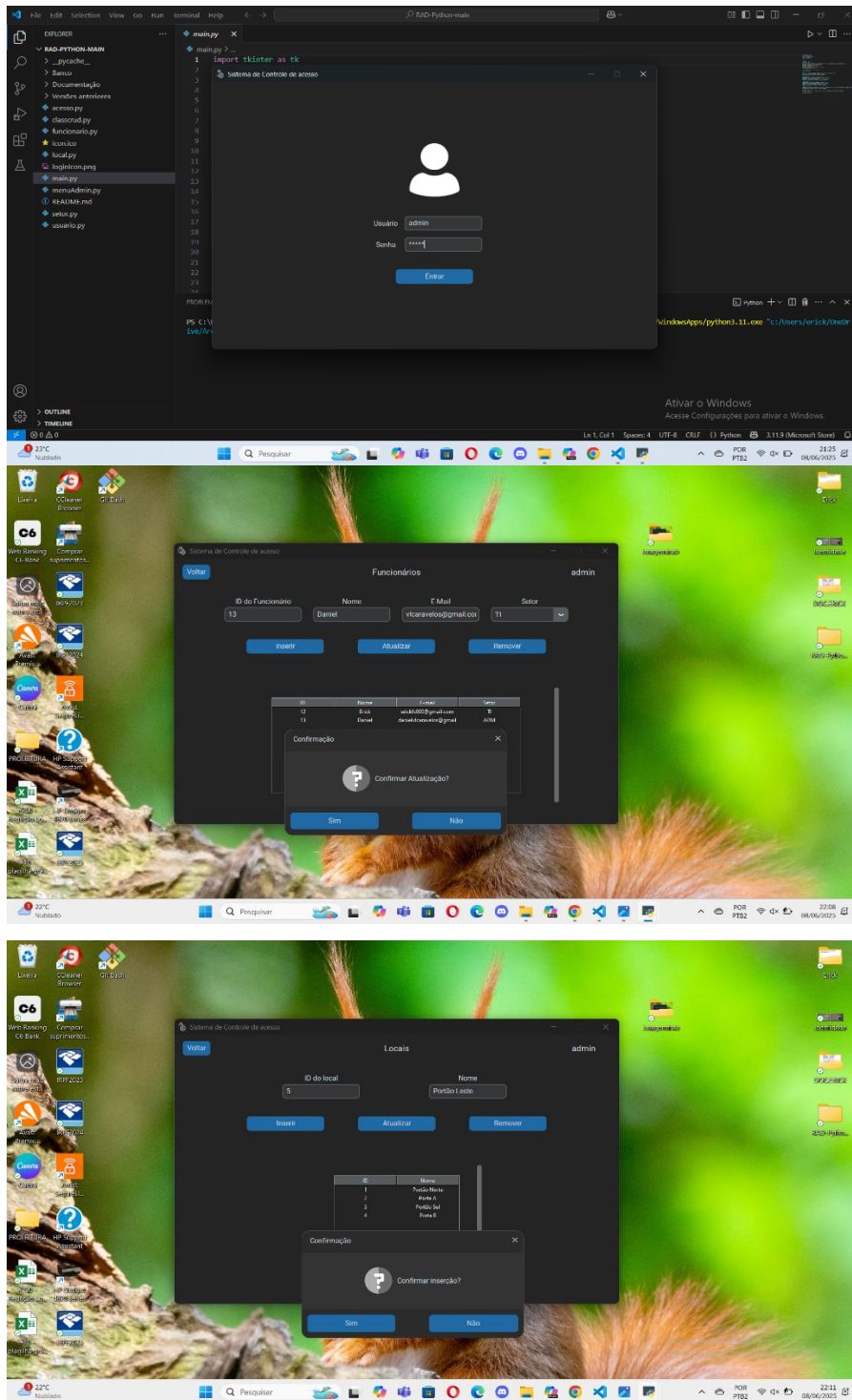


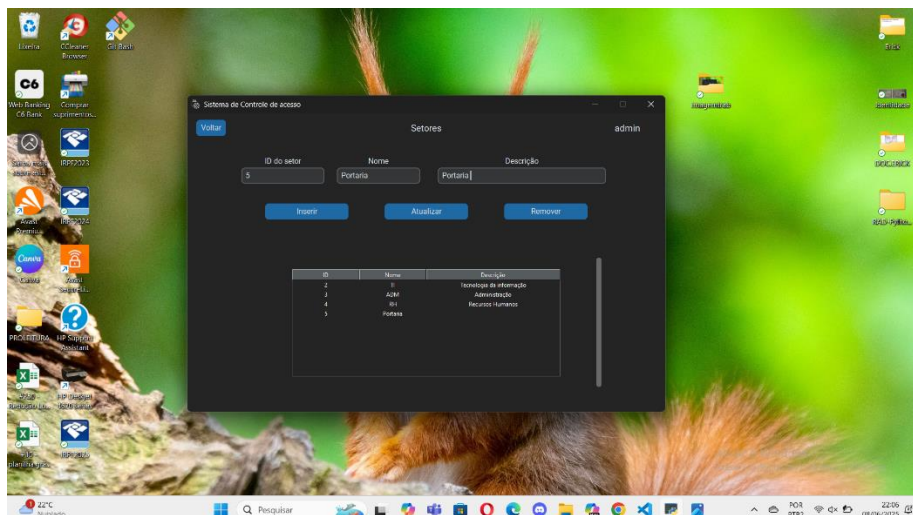
2.5.2. LÓGICO



2.5.3. Sistema em funcionamento







2.6. ITENS

Temos aqui alguns itens descritos em tópicos sobre um programa de entrada e saída em um sistema de segurança desenvolvido utilizando a abordagem de Desenvolvimento Rápido de Aplicações (RAD):

1. Objetivo do Programa:

- Registrar de forma precisa e automatizada a entrada e saída de usuários em um sistema de segurança.

2. Funcionalidades Principais:

- Registro eficiente das entradas e saídas de usuários.
- Captura de informações como data e horário de cada evento.
- Interface intuitiva e amigável para facilitar a utilização do programa pelos usuários.
- Funcionalidades de personalização para adaptar o programa às necessidades específicas de cada organização.

3. Integração e Personalização:

- Integração de funcionalidades que permitem configurar políticas de acesso.
- Possibilidade de definir horários de funcionamento específicos.

4. Segurança e Confiabilidade:

- Adoção de medidas para garantir a segurança dos dados registrados.
- Implementação de protocolos de segurança para proteger informações sensíveis.
- Prevenção de possíveis falhas ou violações de segurança por meio de medidas adequadas.

5. Avaliação e Melhorias Contínuas:

- Realização de testes práticos para avaliar a eficácia do programa.
- Coleta de feedback dos usuários para identificar áreas de melhoria.
- Busca constante por refinamentos e atualizações para futuras versões do programa.

2.7. ATUALIZAÇÃO DAS VERSÕES

- Foi inserido no projeto as funcionalidades de INSERIR e LISTAR as entradas e saídas. (Versão 1).

- Foi criada a classe ACESSO.PY para substituir os objetos ENTRADA e SAÍDA. (Versão 2).

- Foi implementada a versão gráfica da classe ACESSO.PY usando a biblioteca do Python TKINTER. (Versão 3).

- Foi implementada o banco de dados e a interface gráfica de LOCAL, criando a classe LOCAL.PY. (Versão 4).

- Foi implementado o sistema de Login para aumentar a segurança do projeto, criando a rotina LOGIN.PY.

- Foi implementado o banco de dados e a interface gráfica de SETOR, criando a classe SETOR.PY.

- Foi implementado o banco de dados e a interface gráfica de FUNCIONARIO, criando a classe FUNCIONARIO.PY.

- Foi implementado o banco de dados e a interface gráfica de USUARIO, criando a classe USUARIO.PY (Versão 5).

- Foi inserido níveis de acesso para melhorar a segurança do projeto, criando a rotina MENUADMIN.PY, página que tem acesso e controle para inserir, excluir, alterar e consultar as classes SETOR, FUNCIONARIO, USUARIO, LOCAL e ACESSO. (Versão 6).

- Foi inserido a funcionalidade de dar as opções de Administrador para usuários que até então não tinham este poder.

- Alteração e embelezamento da interface gráfica do projeto, usando a biblioteca CUSTOM TKINTER para o auxiliar nas mudanças. (Versão 7).

3. CONCLUSÃO

A implementação bem-sucedida do programa em Python utilizando a abordagem de Desenvolvimento Rápido de Aplicações (RAD) alcançou os objetivos delineados de forma satisfatória. O programa desenvolvido permite o registro preciso e automatizado das entradas e saídas de usuários em ambientes corporativos, capturando informações essenciais como data e horário. Além disso, a interface intuitiva e amigável, juntamente com funcionalidades de personalização, garante uma experiência de uso eficaz e adaptável às necessidades específicas de cada organização. A segurança e confiabilidade dos dados foram priorizadas, com medidas adequadas implementadas para proteger as informações sensíveis. Por meio de testes práticos e feedback dos usuários, a eficácia e usabilidade do programa foram avaliadas, buscando constantemente melhorias para

futuras versões. No geral, essa solução oferece uma gestão de acessos eficiente e robusta, contribuindo significativamente para a segurança, organização e produtividade dos ambientes corporativos.

REFERÊNCIAS

brModelo - Ferramenta de Ensino: Modelagem de Dados (MER). Disponível em: <<http://www.sis4.com/brModelo/download.html>>. Acesso em: 26 mar. 2024.

CUSTOMTKINTER DOCUMENTATION: Documentação oficial do CustomTkinter. Disponível em: <<https://customtkinter.tomschimansky.com/documentation/>>. Acesso em: 1 mai. 2024.

GITHUB: Plataforma líder mundial de desenvolvimento de software, colaboração e segurança. Disponível em: GitHub. Acesso em: 10 mar. 2024.

PROGRAMIZ PYTHON TUTORIAL: Tutorial de Python strftime() - datetime para string. Disponível em: <<https://www.programiz.com/python-programming/datetime/strftime>>. Acesso em: 1 abr. 2024.

PYTHON DOCUMENTATION: Documentação do hashlib — Hashes seguros e resumos de mensagens — Python 3.12.3. Disponível em: <<https://docs.python.org/3/library/hashlib.html>>. Acesso em: 16 abr. 2024.

PYTHON DOCUMENTATION: Documentação do tkinter.ttk — Widgets temáticos Tk. Disponível em: <<https://docs.python.org/3/library/tkinter.ttk.html>>. Acesso em: 12 abr. 2024.

PYTHON DOCUMENTATION: sqlite3 — DB-API 2.0 interface for SQLite databases. Disponível em: <<https://docs.python.org/3/library/sqlite3.html>>. Acesso em: 19 mar. 2024.

VISUAL STUDIO CODE: Editor de código Visual Studio Code. Disponível em: <<https://code.visualstudio.com/Download>>. Acesso em: 10 mar. 2024.

W3SCHOOLS PYTHON TUTORIAL: Tutorial de Python. Disponível em: <<https://www.w3schools.com/python/>>. Acesso em: 15 mar. 2024.