Universidade Estácio de Sá

curso Análise e desenvolvimento de sistemas

UNIDADE NOVA AMÉRICA

**TRABALHO DE Desenvolvimento Rápido de Aplicações em Python**

Rio de Janeiro

Novembro / 2024

**202203669711– Mateus Cabral Gama Oliveira**

**TRABALHO DE Desenvolvimento**

**Rápido de Aplicações**

**em Python**

Trabalho de RAD apresentado a Universidade Estácio de Sá, como exigência para avaliação na disciplina Desenvolvimento Rápido de Aplicação em Python

Orientador:

Prof. Ronaldo Candido dos Santos

SUMÁRIO

[1 INTRODUÇÃO 3](#_Toc84406832)

[1.1 DESCRIÇÃO DO PROBLEMA 3](#_Toc84406833)

[4](#_Toc84406834)

[2 DESENVOLVIMENTO 4](#_Toc84406835)

[2.1 Criação do Banco de Dados 4](#_Toc84406836)

2.2 Validação de telefone........................................................................................................................5

2.3 Consulta de Clientes..........................................................................................................................5

2.4 Cadastro Novo Cliente.......................................................................................................................5

2.5 Atualização de Cortes.........................................................................................................................5

2.6 Geração de Relatório..........................................................................................................................5

3 CONCLUSÃO...........................................................................................................................................6

4 [REFERÊNCIAS 6](#_Toc84406838)

# INTRODUÇÃO

## DESCRIÇÃO DO PROBLEMA

Atualmente, muitas barbearias enfrentam dificuldades para gerenciar suas operações diárias de forma eficaz, especialmente no que diz respeito ao controle de informações de clientes, serviços e agendamentos. A utilização de métodos manuais, como planilhas ou registros em papel, pode resultar em erros, perda de dados e ineficiência, tornando o processo de gestão desorganizado e propenso a falhas. A falta de um sistema digital centralizado para administrar o cadastro de clientes, o histórico de serviços e o agendamento de cortes de cabelo, barbas e outros tratamentos pode impactar diretamente a produtividade e a qualidade do atendimento ao cliente. Portanto, há uma necessidade evidente de desenvolver uma solução tecnológica para facilitar essas operações e melhorar a gestão da barbearia, resultando em um ambiente de trabalho mais organizado e eficiente.

Para atender a essa necessidade, propomos o desenvolvimento de uma aplicação utilizando a metodologia de Rapid Application Development (RAD) com a linguagem de programação Python. O RAD é uma abordagem de desenvolvimento de software que enfatiza a rápida prototipagem e iteração, permitindo que as aplicações sejam desenvolvidas de forma mais ágil e adaptável às mudanças de requisitos. Com Python, uma linguagem conhecida por sua simplicidade e versatilidade, é possível criar uma aplicação robusta e eficiente para o gerenciamento de barbearias.

A aplicação proposta incluirá funcionalidades essenciais como o cadastro de clientes, onde será possível armazenar informações detalhadas sobre cada cliente, incluindo histórico de serviços e preferências. Além disso, o sistema permitirá o agendamento de serviços, facilitando a organização dos horários e evitando conflitos de agenda. A integração de um banco de dados centralizado garantirá que todas as informações sejam armazenadas de forma segura e acessível, minimizando o risco de perda de dados.

Com essa solução tecnológica, as barbearias poderão melhorar significativamente sua eficiência operacional, proporcionando um atendimento mais ágil e personalizado aos clientes. A implementação de um sistema digital não só otimiza o tempo dos funcionários, mas também melhora a experiência do cliente, resultando em maior satisfação e fidelização.

## OBJETIVOS

O principal objetivo deste projeto é desenvolver um sistema de gerenciamento para uma barbearia, utilizando a linguagem de programação Python, que possibilite:

* Cadastrar e gerenciar informações de clientes de forma prática e centralizada.
* Agendar e controlar serviços oferecidos, como cortes de cabelo, barbas
* Otimizar o tempo e os recursos da barbearia, proporcionando uma interface intuitiva e funcional para o usuário.
* Garantir a segurança e integridade dos dados, através da utilização de um banco de dados confiável (SQLite).
* Melhorar a organização e o atendimento ao cliente, automatizando processos que, de outra forma, seriam manuais e propensos a erros.

# DESENVOLVIMENTO

O desenvolvimento deste projeto visa criar um sistema de gerenciamento para uma barbearia, utilizando a linguagem de programação Python e a metodologia de Rapid Application Development (RAD). O objetivo é proporcionar uma solução eficiente e prática para o cadastro de clientes, agendamento de serviços e controle de histórico de cortes, melhorando a organização e a produtividade da barbearia.

Para a implementação do sistema, utilizamos a linguagem de programação Python devido à sua simplicidade e versatilidade. A biblioteca Tkinter foi escolhida para a criação da interface gráfica, proporcionando uma experiência de usuário intuitiva e funcional. O banco de dados SQLite foi selecionado para armazenar as informações de clientes e serviços, garantindo a segurança e integridade dos dados.

## Criação do Banco de dados

A função criar\_banco é responsável por criar as tabelas necessárias no banco de dados SQLite. São criadas duas tabelas: clientes para armazenar informações dos clientes e cortes para registrar as datas dos cortes realizados.

## Validação de telefone

A função “validar\_telefone” verifica se o número de telefone inserido é válido, garantindo que apenas números sejam aceitos e que o comprimento mínimo seja respeitado.

## Consulta de Cliente

A função “consultar\_cliente” permite buscar informações de um cliente no banco de dados a partir do número de telefone, facilitando a verificação de clientes já cadastrados

## Cadastro de Novo Cliente

A função “cadastrar\_cliente” insere um novo cliente no banco de dados, registrando também a data do primeiro corte, se aplicável. Em caso de duplicidade de telefone, uma mensagem de erro é exibida.

## Atualização de Cortes

A função “atualizar\_cortes” atualiza o número de cortes realizados por um cliente e registra a data do novo corte no banco de dados.

## Geração de Relatório

A função “abrir\_relatorio” gera um relatório com a lista de todos os clientes cadastrados, exibindo nome, telefone e número de cortes realizados. O relatório é apresentado em uma nova janela criada com Tkinter.

# CONCLUSÃO

A implementação do sistema de gerenciamento para a barbearia automatizou processos manuais, resultando em uma significativa economia de tempo e redução de erros. A rápida consulta ao histórico de serviços permitiu um atendimento mais personalizado e eficiente, aumentando a satisfação dos clientes. A utilização do banco de dados SQLite garantiu a integridade e segurança das informações armazenadas, minimizando o risco de perda de dados. A interface gráfica desenvolvida com Tkinter proporcionou uma experiência de usuário amigável, facilitando o uso do sistema pelos funcionários da barbearia.

Identificou-se a necessidade de aprimorar a interface gráfica, incluir relatórios mais detalhados e integrar o sistema com outras plataformas, como sistemas de pagamento e marketing digital. Essas conclusões destacam os principais benefícios e áreas de melhoria do sistema de gerenciamento desenvolvido, demonstrando sua eficácia na organização e produtividade da barbearia.

4 REFERÊNCIAS

FLOWUP. Rapid Application Delivery (RAD): O que é e pra que serve? Disponível em: <https://www.flowup.me/blog/rapid-application-delivery-rad/>. Acesso em: 29 out. 2024

PYTHONCLUB. Guia rápido de comandos SQLite3. Disponível em: <https://pythonclub.com.br/guia-rapido-comandos-sqlite3.html>. Acesso em: 29 out. 2024

PYTHON SOFTWARE FOUNDATION. tkinter — Interface Python para Tcl/Tk. Disponível em: https://docs.python.org/pt-br/3/library/tkinter.html. Acesso em: 29 out. 2024.