

Aluno: Manuel Marcelo R. Varas

RGM: 40817270

Projeto: Projeto Integrador Transdisciplinar em Banco de Dados I

1. Introdução

Este projeto, intitulado "Projeto Integrador Transdisciplinar em Banco de Dados I", foi desenvolvido no âmbito do curso Técnico Superior em Banco de Dados da Cruzeiro do Sul Virtual. O projeto aborda a Situação-Problema 2, que envolve a criação de um sistema de banco de dados para um consultório médico. O sistema visa otimizar a gestão de informações cruciais como pacientes, médicos, consultas, exames, convênios e aspectos financeiros, garantindo segurança e eficiência no manuseio dos dados.

Além da implementação do banco de dados, um esforço adicional foi dedicado à criação de interfaces gráficas em Python para proporcionar uma interação mais amigável e funcional com o sistema.

2. Objetivos

O principal objetivo deste projeto é desenvolver um sistema de banco de dados centralizado e seguro para o consultório médico. Este sistema tem como finalidade:

- Centralizar as informações do consultório em um único local, facilitando o acesso e a gestão dos dados.
- Proteger os dados contra perdas, através de backups automatizados e medidas de segurança.
- Agilizar processos administrativos, como agendamentos de consultas, gestão de prontuários eletrônicos e faturamento.
- Garantir a conformidade com a Lei Geral de Proteção de Dados (LGPD), assegurando a privacidade e a segurança das informações dos pacientes.
- Adicionalmente, criar uma interface de usuário intuitiva para facilitar a interação com o banco de dados, tornando o sistema mais acessível e prático para os usuários do consultório.

3. Revisão de Conceitos

Para o desenvolvimento deste projeto, foram aplicados conhecimentos nas seguintes áreas:

- **Modelagem Relacional (MySQL):** Utilização do MySQL para a criação e gestão do banco de dados, definindo as tabelas e seus relacionamentos.
- **Normalização de Tabelas:** Aplicação das formas normais (até 3FN) para reduzir a redundância e a dependência dos dados, garantindo a integridade do banco de dados.
- **Criação de Constraints (PK/FK):** Definição de chaves primárias (PK) e chaves estrangeiras (FK) para estabelecer e manter os relacionamentos entre as tabelas.
- **Utilização de ENUMs:** Implementação de tipos ENUM para validar os dados inseridos nas tabelas, assegurando a consistência das informações.
- **Backup Automatizado (AWS S3):** Configuração de backups automatizados na nuvem utilizando o serviço AWS S3 para garantir a recuperação dos dados em caso de falhas.
- **Segurança de Dados:** Aplicação de medidas de segurança para proteger as informações dos pacientes e do consultório contra acessos não autorizados.
- **Integração MySQL com Python:** Utilização de bibliotecas Python (como `mysql.connector` e `tkinter`) para conectar o banco de dados MySQL com as interfaces gráficas, permitindo a manipulação dos dados de forma interativa.

4. Metodologia

A metodologia adotada para a resolução do problema seguiu uma abordagem em camadas, dividida em três fases principais:

- **Modelagem Conceitual:** Identificação das entidades relevantes (pacientes, médicos, consultas, etc.) e seus relacionamentos, representada pelo Diagrama Entidade-Relacionamento (DER) (Anexo 1).
- **Modelagem Lógica:** Definição das tabelas, colunas, tipos de dados e chaves primárias e estrangeiras no MySQL, normalizando as tabelas até a 3ª Forma Normal (3FN).
- **Modelagem Física:** Implementação do banco de dados no SGBD MySQL, criação das tabelas, constraints e aplicação de técnicas de otimização de performance, como indexação.
- **Desenvolvimento das Interfaces em Python:** Criação de interfaces gráficas utilizando a biblioteca `tkinter` para permitir a interação dos usuários com o banco de dados, facilitando a inserção, consulta, atualização e exclusão de dados.

5. Cronograma

O desenvolvimento do projeto foi planejado e executado no seguinte cronograma:

- **Modelagem (Conceitual e Lógica):** 3 dias úteis
- **Implementação SQL (Modelagem Física):** 2 dias úteis
- **População com Dados de Teste:** 1 dia útil
- **Configuração de Backups na AWS:** 1 dia útil
- **Desenvolvimento das Interfaces em Python:** 3 dias úteis (Este é um tempo adicional ao projeto original)
- **Testes e Ajustes:** 2 dias úteis
- **Tempo Total:** 12 dias úteis

6. Procedimentos e Materiais Utilizados

As etapas e ferramentas utilizadas para a implementação do sistema foram:

- **Levantamento de Requisitos:** Entrevistas com os stakeholders (médicos e administradores do consultório) para identificar as necessidades e os requisitos do sistema.
- **Modelagem no BrModelo:** Utilização da ferramenta BrModelo para a criação do Diagrama Entidade-Relacionamento (DER), representando graficamente as entidades e seus relacionamentos (Anexo 1).
- **Implementação em MySQL:** Criação do banco de dados e das tabelas no MySQL, utilizando o script SQL fornecido (Anexo 2).
- **Criação de Usuários e Permissões:** Configuração dos usuários e suas permissões de acesso ao banco de dados utilizando o comando GRANT do MySQL, garantindo a segurança das informações.
- **Automatização de Backups:** Implementação de backups automatizados utilizando a AWS CLI (Command Line Interface) e os comandos `mysqldump` para exportar o banco de dados e S3 para armazená-lo na nuvem.
- **Desenvolvimento de Queries Críticas:** Criação de consultas SQL otimizadas para gerar relatórios de consultas, pagamentos e outras informações relevantes para a gestão do consultório.
- **Desenvolvimento das Interfaces Gráficas:**
 - Utilização da biblioteca `tkinter` do Python para criar as janelas, formulários e elementos de interação com o usuário.
 - Utilização da biblioteca `mysql.connector` para estabelecer a conexão entre as interfaces Python e o banco de dados MySQL, permitindo a execução de comandos SQL e a recuperação dos dados.
 - **Observação Importante:** Para executar as interfaces em Python, é necessário verificar e, se necessário, alterar a senha de conexão com o banco de dados nas variáveis `password` dentro dos arquivos Python (por

exemplo, `Agenda.py`, `Consulta.py`, etc.). A senha padrão utilizada no código é "M@taturu.1981".

- **Códigos Fonte:** Os códigos fonte do sistema, incluindo os scripts SQL (Anexo 2) e os códigos das interfaces em Python (Anexo 3).

7. Resultados e Discussão

O projeto resultou em um sistema de banco de dados funcional e eficiente, que atende aos requisitos iniciais do consultório médico, e que foi significativamente aprimorado pela adição das interfaces em Python. As principais entregas do projeto foram:

- Um banco de dados com 10 tabelas inter-relacionadas, modelado para armazenar e gerenciar as informações do consultório de forma organizada.
- Backups automatizados diários na AWS, garantindo a segurança e a disponibilidade dos dados.
- Consultas SQL otimizadas para a geração de relatórios gerenciais, facilitando a tomada de decisões.
- Interfaces gráficas intuitivas em Python que permitem aos usuários interagir com o banco de dados de forma fácil e eficiente, realizando operações como cadastro de pacientes, agendamento de consultas, emissão de relatórios, etc.

A integração do MySQL com Python demonstrou a capacidade de transformar um banco de dados em uma ferramenta verdadeiramente útil para o dia a dia do consultório, proporcionando uma experiência de usuário mais completa e amigável.

8. Melhorias Futuras

Para aprimorar ainda mais o sistema, algumas melhorias futuras são sugeridas:

- Integração com um front-end web/mobile para facilitar o acesso e a utilização do sistema pelos usuários em diferentes plataformas.
- Implementação de triggers para auditoria, registrando as alterações realizadas no banco de dados para garantir a rastreabilidade das informações.
- Migração para o Amazon Aurora em caso de necessidade de escalabilidade, oferecendo maior desempenho e disponibilidade para o sistema.
- Aprimoramento das interfaces em Python, adicionando mais funcionalidades e melhorando o design para tornar a experiência do usuário ainda mais agradável.

9. Conclusão

Este projeto "Projeto Integrador Transdisciplinar em Banco de Dados I", desenvolvido no curso Técnico Superior em Banco de Dados da Cruzeiro do Sul Virtual, demonstrou a aplicação prática dos conhecimentos adquiridos para resolver um problema real. O

sistema de banco de dados desenvolvido contribui para a organização, segurança e eficiência da gestão de informações em um consultório médico, otimizando os processos administrativos e melhorando o atendimento aos pacientes.

A implementação das interfaces em Python representou um valor agregado significativo ao projeto, evidenciando a importância da integração entre o banco de dados e a camada de apresentação para a criação de um sistema completo e funcional.

Anexos

- Anexo 1: Diagrama Entidade-Relacionamento (DER)
- Anexo 2: Script SQL para criação do banco de dados
- Anexo 3: Códigos fonte (interfaces em Python)

Incluí um aviso bem claro sobre a necessidade de verificar e, possivelmente, alterar a senha nos arquivos Python para garantir o funcionamento correto das interfaces. Espero que isso ajude a evitar problemas na execução do projeto!