

Documentação das Estratégias do Código

1. Objetivo:

O objetivo do código é realizar a migração de dados fictícios de um sistema legado para a base de dados de uma clínica fictícia chamada MedicalChallenge. As etapas para isso incluem a instalação do MariaDB, restauração do banco de dados MedicalChallenge, migração dos dados do sistema legado para o banco temporário, e geração de um dump dos dados migrados para o banco da MedicalChallenge.

2. Ambiente de Desenvolvimento:

O ambiente de desenvolvimento está configurado utilizando containers Docker para fornecer um ambiente virtual isolado e portátil. As principais ferramentas e dependências utilizadas são:

PHP 8.1-fpm: A imagem base para o ambiente PHP.

MariaDB: Banco de dados relacional usado para a base de dados MedicalChallenge e um banco temporário.

Redis: Sistema de armazenamento em cache usado como um serviço adicional.

3. Dockerfile:

O Dockerfile define as instruções para construir a imagem do contêiner PHP. Ele inclui:

- Instalação de dependências do sistema, como Git, Curl, e Mariadb-client.
- Configuração de extensões PHP necessárias para o projeto, como mysqli, pdo_mysql, mbstring, etc.
- Instalação do Composer e definição de um usuário não privilegiado para executar comandos Composer e Artisan (somente se necessário em alguma etapa do projeto).
- Configuração do Redis como um serviço adicional.

4. Docker Compose:

O Docker Compose é usado para orquestrar vários serviços relacionados ao projeto. Ele define os serviços e suas configurações, como volumes, redes, e dependências.

Serviço PHP: Baseado no Dockerfile, monta o diretório de trabalho e depende do serviço Redis.

Serviço DB (MariaDB): Fornece o banco de dados principal para o MedicalChallenge.

Serviço Mariadb: Banco de dados temporário usado durante a migração.

Serviço Redis: Banco de dados chave-valor usado como um serviço adicional para armazenamento em cache.

5. Funcionalidades Principais:

O código “migrate.php” inclui as seguintes funcionalidades principais:

5.1. Conexão com os Bancos de Dados:

Estabelece conexão com o banco de dados da clínica fictícia MedicalChallenge, e também com o banco temporário utilizado na migração.

5.2. Leitura e Verificação dos Arquivos CSV:

Verifica se os arquivos contendo os dados do sistema legado têm a extensão .csv.

5.3. Migração dos Dados:

Realiza a migração dos dados dos pacientes e agendamentos para o banco de dados MedicalChallenge. Os dados são tratados e inseridos nas tabelas correspondentes no banco de dados. Também verifica se os registros já existem no banco de dados antes de inserir novos registros.

5.4. Geração de Dump dos Dados Migrados:

Gera um dump dos dados já migrados para o banco de dados MedicalChallenge. O dump é salvo em um arquivo chamado 'medicalChallenge.sql'.

6. Estratégias Implementadas:

6.1. Modularização do Código:

O código foi modularizado em funções para facilitar a manutenção e reutilização. As funções foram agrupadas em uma biblioteca separada “lib.php”.

6.2. Verificação de Extensão de Arquivo:

Antes de processar os arquivos CSV, o código verifica se eles têm a extensão .csv. Isso ajuda a garantir que apenas arquivos no formato esperado sejam processados.

6.3. Tratamento de Dados:

Os dados dos arquivos CSV são tratados adequadamente antes de serem inseridos no banco de dados, como por exemplo, a conversão de codificação de caracteres para UTF-8, quando necessário.

6.4. Verificação de Duplicados:

Antes de inserir novos registros no banco de dados, o código verifica se já existem registros idênticos, evitando a inserção de dados duplicados na base de dados.

6.5. Geração de Dump:

Após a migração dos dados, o código gera um dump dos dados migrados para facilitar a posterior restauração ou backup dos dados.

7. Conclusão:

O código implementa de forma eficiente as estratégias necessárias para realizar a migração dos dados do sistema legado para o banco de dados da clínica fictícia MedicalChallenge, garantindo consistência e integridade dos dados durante todo o processo. A modularização do código, a verificação cuidadosa dos dados e a geração de dump fornecem uma solução robusta e escalável para o desafio proposto.