

UNIFACS (CAMPUS PARALELA)

Murilo Lima Santos - 12722129125

Guilherme Reis Correia - 12722123056

Petson Brenner Santos Alves - 12722130640

Davi de Sena Barbosa Silva – 12722130689

Adelirio Alexandre Dos Santos Neto - 12722127337

RELATÓRIO DE PROJETO A3 (UC: SISTEMAS DISTRIBUIDOS)

SALVADOR – BA

2023

Descrição dos requerimentos de softwares necessários para execução da aplicação

Passo 1: Baixar o Python:

Acesse o site oficial do Python em <https://www.python.org/> e faça o "Download" no menu principal.

Passo 2: Instalar o Python:

Após o download do instalador, execute o arquivo baixado. • Marcando a opção "Add Python to PATH" durante a instalação para facilitar o uso do Python no prompt de comando. Clique em "Install Now" para iniciar a instalação.

Passo 3: Verificar a instalação:

Abra o terminal (ou prompt de comando no Windows) e digite o seguinte comando para verificar se o Python foi instalado corretamente e para ver a versão instalada: `python --version` Agora, está pronto para começar a usar o Python no computador. Podendo executar scripts Python, iniciar o interpretador interativo e muito mais.

Passo 4: Instalar o MySQL:

Faça o download do MySQL Workbench: Acesse o site oficial do MySQL para baixar a versão mais recente do MySQL Workbench. O link é <https://dev.mysql.com/downloads/workbench/>.

•

Escolha o Sistema Operacional: Selecione o sistema operacional que você está usando (Windows, macOS ou Linux) e clique no botão de download correspondente.

Inicie o Instalador: Execute o arquivo de instalação que você baixou. Uma tela de boas-vindas fornece várias opções. Escolha a primeira opção: Install MySQL Products.

Faça o download dos mais recentes produtos MySQL: O instalador do MySQL verifica e baixa os mais recentes produtos MySQL, incluindo servidor MySQL, MySQL Workbench.

Existem vários tipos de configuração disponíveis. Escolha a opção **Full** para instalar todos os produtos e recursos do MySQL.

Clique no botão **Next** para configurar o MySQL Database Server, escolha o tipo de configuração e porta MySQL (3306 por padrão) e clique no botão **Next** para continuar.

Escolha uma senha para a conta root. Anote a senha e mantenha-a em segurança se estiver instalando o servidor de banco de dados MySQL em um servidor de produção.

Escolha os detalhes do serviço do Windows, incluindo o Nome do serviço do Windows e o tipo de conta, depois clique no botão **Next** para continuar.

O Instalador do MySQL está configurando o servidor de banco de dados MySQL. Aguarde até terminar e clique no botão **Next** para continuar.

O Instalador do MySQL instala bancos de dados e modelos de amostra.

A instalação está concluída. Clique no botão **Finish** para fechar o assistente de instalação e iniciar o MySQL Workbench.

Passo 5: Instalar o VS Code:

A instalação no Windows é relativamente simples, basta acessar o [site oficial do Visual Studio Code](#) e fazer o Download da sua última versão, clicando no botão "Download for Windows".

Em seguida, execute o instalador baixado e siga as instruções na tela, não esqueça de marcar a opção "Add to Path" para adicionar o Visual Studio Code nas variáveis de ambiente.

Finalizada a instalação, feche todos os terminais abertos e abra um novo (Power Shell ou [Windows Terminal](#)) e digite o comando abaixo.

```
code --version
```

Se tudo ocorrer como esperado, você verá a versão do Visual Studio Code instalada em sua máquina e estará tudo pronto para prosseguir.

Justificativa da escolha da tecnologia

A ideia foi criar um servidor web capaz de enviar informações de maneira segura, eficiente e padronizada. Diante dos requisitos do projeto, a opção escolhida foi a utilização de uma API própria, implementada em Python, uma linguagem de fácil desenvolvimento.

O principal diferencial do projeto é a adoção do Flask como framework. O Flask é conhecido por sua simplicidade no desenvolvimento de rotas, tratamento de requisições, protocolos e pela facilidade de integração com o frontend por meio da linguagem Jinja, que é incorporada ao HTML de forma simples e eficiente.

Instruções para instalação e execução da aplicação

Passo 1: Clonar o Repositório

Comece clonando o repositório da entrega A3. Para fazer isso, acesse o GitHub, vá até o repositório desejado e clique na aba "CODE". Copie o link do repositório. Em seguida, escolha a pasta onde deseja armazenar o projeto. Na barra de caminhos, substitua o caminho atual por "cmd" e pressione Enter. Agora, digite o comando `git clone` seguido pelo link do repositório que você copiou.

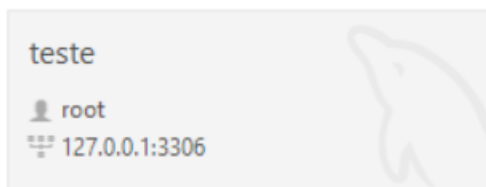


Passo 2: Criar um Banco de Dados

Abra o MySQL Workbench e acesse a sua local instance:

Local instance:

MySQL Connections + -



Crie um novo banco de dados utilizando os comandos:

```
CREATE DATABASE IF NOT EXISTS nome_banco_de_dados;
```

```
USE nome_banco_de_dados;
```

OBS: Para executar esses comandos, selecione as linhas desejadas e pressione Ctrl + Enter.

Passo 3: Instalar Framework e Conexões

Abra o prompt de comando e insira os seguintes comandos para instalar as dependências necessárias:

```
pip install flask
```

```
pip install flask_restful
```

```
pip install mysql.connector-python
```

Passo 4: Configurar e Executar o Projeto

Abra o Visual Studio Code (VS Code) e vá até a pasta onde o projeto foi clonado. Abra o arquivo `app.py` e localize as linhas de 12 a 15, onde você deve substituir as frases pelos valores de configuração do MySQL:

```
11 db_config = {  
12     'host': 'seu host',  
13     'user': 'seu user',  
14     'password': 'Sua senha',  
15     'database': 'Nome do seu banco de dados'  
16 }
```

Preencha esses requisitos com suas configurações específicas. Após preencher, execute o `arquivo.py`.

Um endereço do menu principal do site será exibido. Selecione o endereço completo e pressione `Ctrl + Click` para acessar o site.

```
WARNING: This is a development server. Do not use it in a production deployment. Use a production WSGI server instead.  
* Running on http://127.0.0.1:5000  
Press CTRL+C to quit  
* Restarting with stat  
* Debugger is active!  
* Debugger PIN: 924-774-948
```

Apresentação e detalhamento sobre a arquitetura, estratégia e algoritmos utilizados

O código está organizado em diferentes seções para lidar com diferentes aspectos do aplicativo, como rotas, manipulação de banco de dados e recursos da API.

Conexão com o Banco de Dados:

A função `conectar_bd` é uma maneira eficaz de gerenciar a conexão com o banco de dados MySQL.

Criação de Tabelas e Inserção de Dados:

A função `criar_tabelas` é responsável por criar as tabelas necessárias no banco de dados e inserir alguns dados iniciais.

Rotas:

As rotas foram definidas, e os métodos HTTP (GET e POST) são utilizados adequadamente para diferentes tipos de operações.

Manipulação do Banco de Dados nas Rotas: O código realiza operações de leitura e escrita no banco de dados de acordo com as operações nas rotas.

API RESTFUL:

No código, foi criada uma classe chamada `ProdutosResource`, que herda da classe `Resource` do Flask-RESTful. Flask-RESTful é uma extensão do Flask que simplifica a criação de APIs RESTful.

A classe `ProdutosResource` define um recurso específico, neste caso, relacionado aos produtos. A implementação atual inclui apenas um método: `get(self)`. Esse método é chamado quando uma requisição HTTP GET é feita para o endpoint associado a esta classe.

O objetivo dessa implementação é fornecer uma representação JSON de todos os produtos no banco de dados quando um cliente faz uma requisição GET.

Dentro desse método, é estabelecida uma conexão com o banco de dados, são executadas consultas SQL para obter todos os produtos e, em seguida, os resultados são convertidos em um formato JSON.

Flask-RESTful:

O uso do Flask-RESTful simplifica a criação de APIs RESTful em comparação com o Flask padrão. Ele fornece abstrações que facilitam a definição de recursos e a manipulação de solicitações HTTP.

JSONify:

A função jsonify do Flask é usada para converter os resultados da consulta em uma resposta JSON. pois facilita o retorno de dados no formato padrão para aplicativos que consomem a API.

