**Nome: João Victor Medeiros Rocha**

**RELATÓRIO SEMANA 09**

**Professor: Éder Alves de Moura**

**Uberlândia**

**2022**

# Roteiro de Atividades

Crie uma pasta em seu repositório GitHub, denominada ‘Semana09’. Nela você desenvolverá

uma aplicação web utilizando o conjunto Python+Flask no Backend (*server side*).

1. Crie uma subpasta ‘web’ na pasta ‘Semana09’ e desenvolva as atividades que estão apresentadas no vídeo:

<https://www.youtube.com/watch?v=Z1RJmh_OqeA>

Este vídeo apresenta o desenvolvimento de um servidor web com o framework Flask, que utiliza a linguagem Python para a criação de páginas dinâmicas.

**PREPARAÇÃO DO AMBIENTE DE DESENVOLVIMENTO**

Para a realização da atividade, vamos utilizar a linguagem de programação Python na sua versão 3.8 em conjunto com o editor de texto Visual Studio Code (VSCode).

Para que possamos iniciar o desenvolvimento é necessário realizar a criação **de uma pasta chamada semana 09** pasta que irá armazenar todos os códigos do nosso projeto. Posteriormente será **criada uma pasta we**b onde iremos armazenar todos os arquivos como por exemplo index.html , css, e o main.py que guardará o nosso servidor flask. Por fim como iremos criar um back-end utilizando o Flask para criar nossos end points , será necessário instalar a biblioteca do Flask e para isso vamos **utilizar o gerenciador de pacotes pip através do seguinte comando: pip install Flask para que possamos ter a biblioteca Flask** . Com a bilbioteca em mãos é possivel começar a criar nossas APIs e seus end points e dar ínicio ao projeto proposto. Veja abaixo um exemplo:

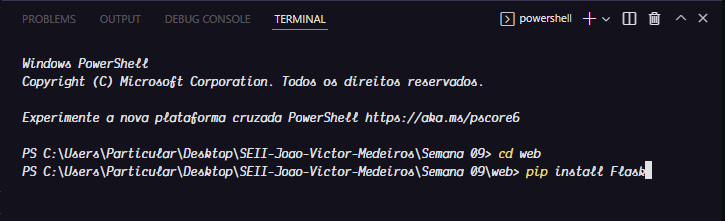


Figura 1: Configuração do ambiente de desenvolvimento.

**CODIFICAÇÃO E DESENVOLVIMENTO DO PROJETO**

Realizado os passos do item anterior partimos para o desenvolvimento do nosso Back-End com Python para que tenhamos os end points prontos para poder lançar a nossa página web para este end point em Flask através de requisições HTTP do tipo: Get , Post , Put , Delete utilizando o modelo CRUD.

Sendo assim foi criado o arquivo app.py veja abaixo:

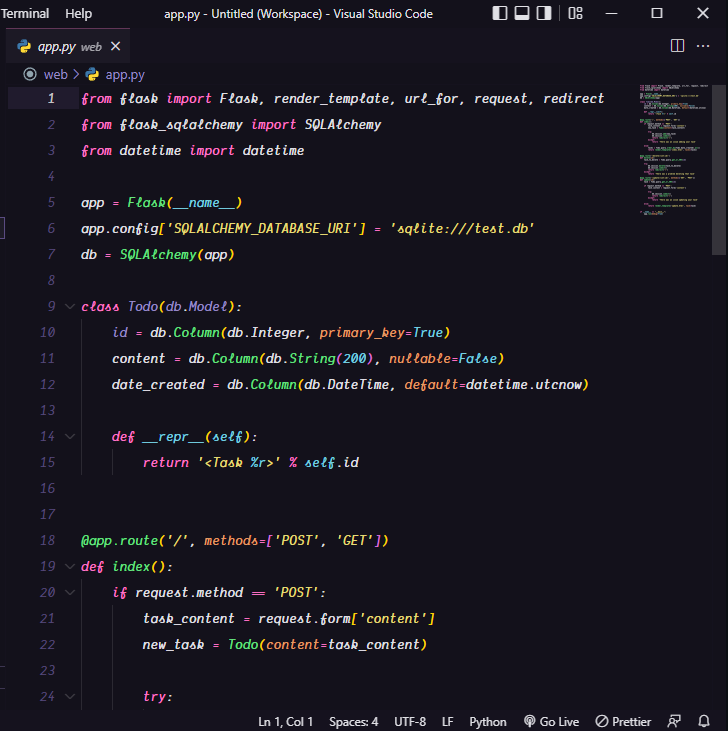


Figura 2: Desenvolvimento do back-end com Flask

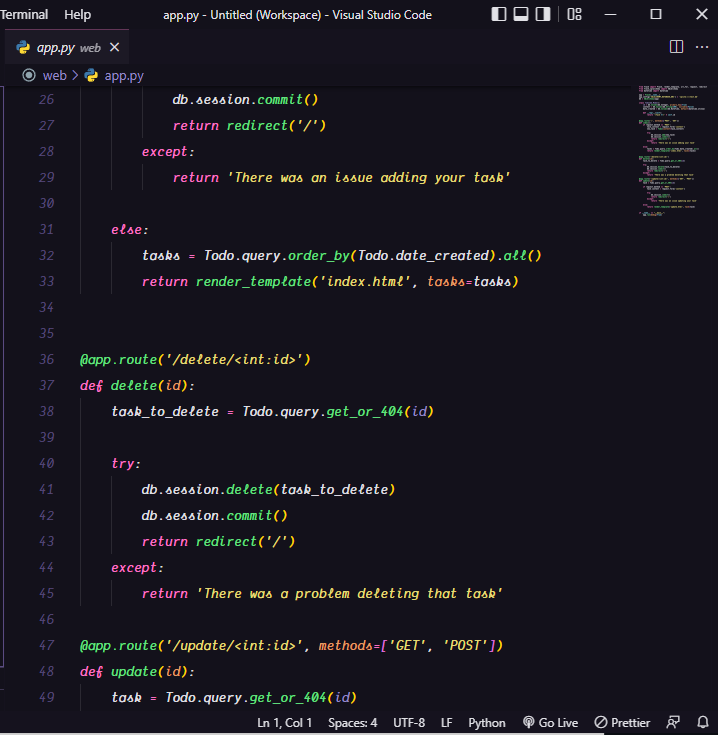


Figura 3: Desenvolvimento do back-end com Flask, segunda parte do código.

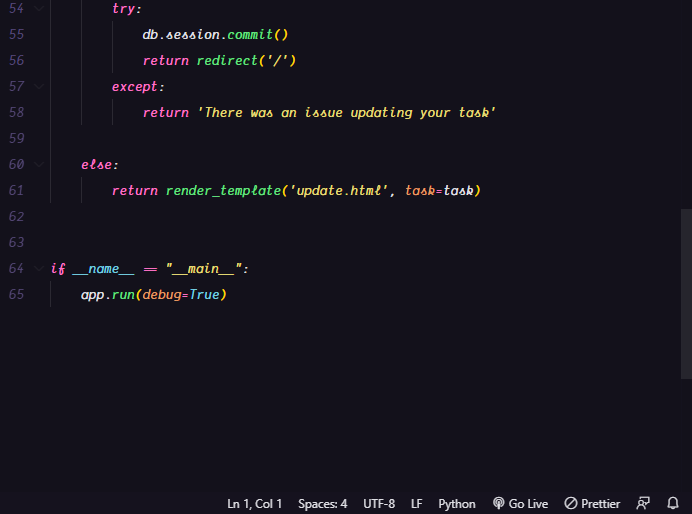


Figura 4: Desenvolvimento do back-end com Flask, terceira parte do código.

Feito o Back-end da nossa aplicação, agora o próximo passo é realizar a criação do nosso cliente que irá consumir de forma REST a APIs dos end points criados com Flask, e para tal iremos utilizar a antiga e conhecida linguagem de marcação HTML , além de utilizarmos a linguagem de estilização ou de folhas de estilo CSS3 que irá trazer para nosso site um “melhor visual”. Logo teremos os seguintes códigos:

* **Códigos HTML:**

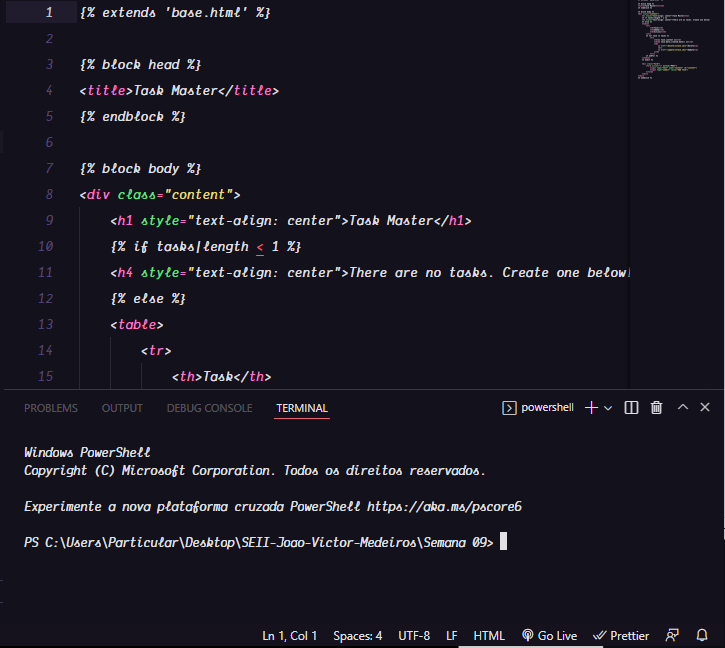


Figura 5: Desenvolvimento do front-end com HTML, primeira parte do código index.html.

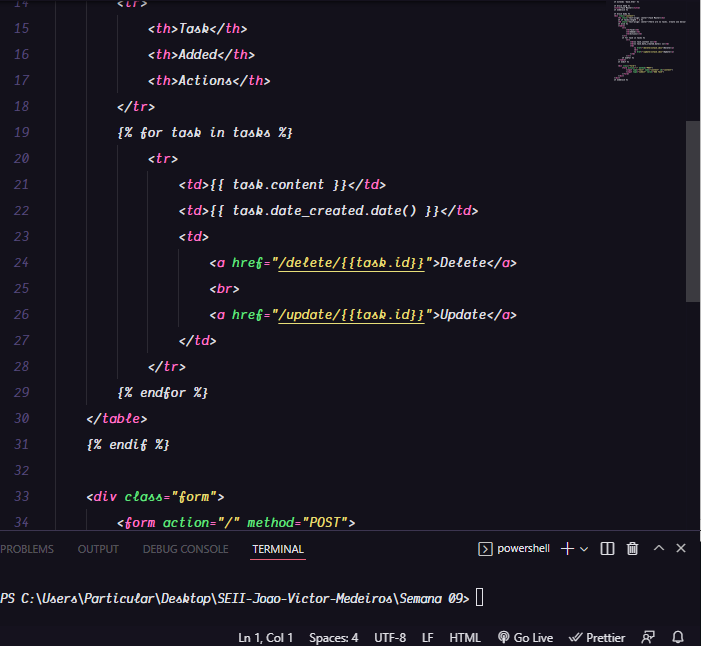


Figura 6: Desenvolvimento do front-end com HTML, segunda parte do código index.html.

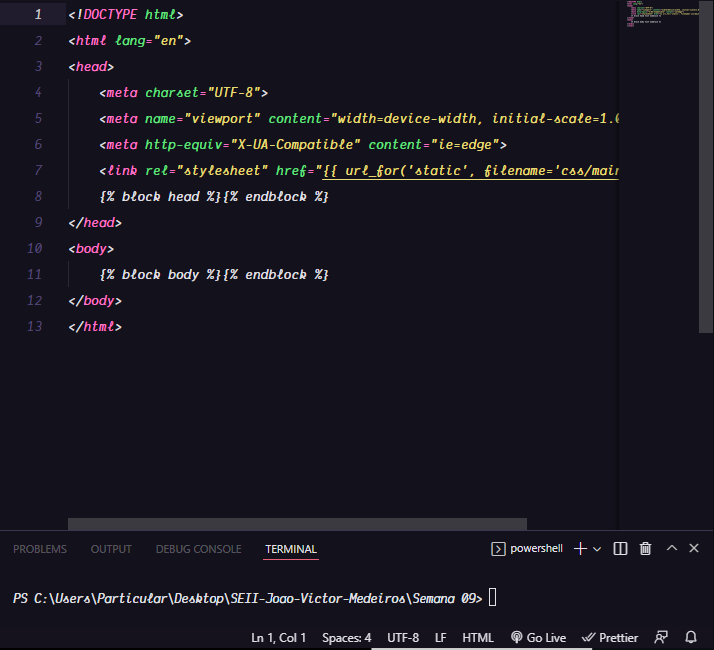


Figura 7: Desenvolvimento do front-end com HTML, código base.html.

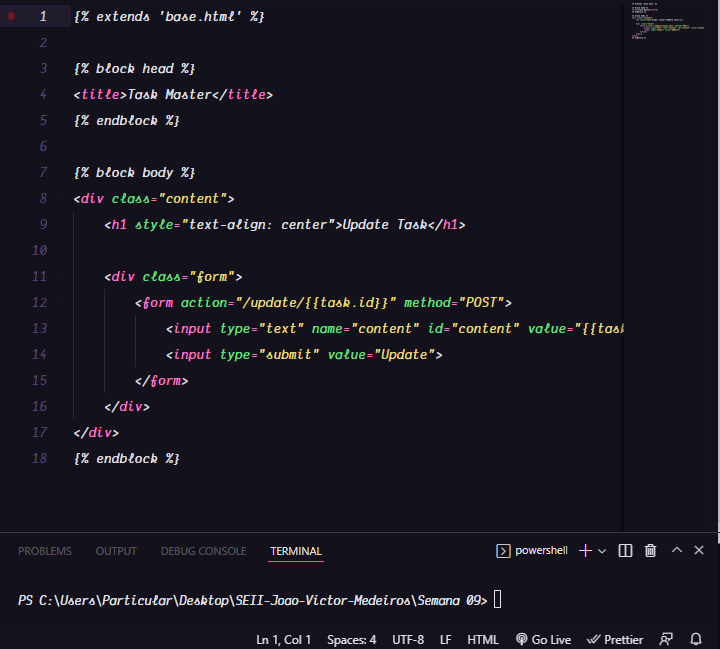


Figura 8: Desenvolvimento do front-end com HTML, código update.html.

* **Código CSS:**

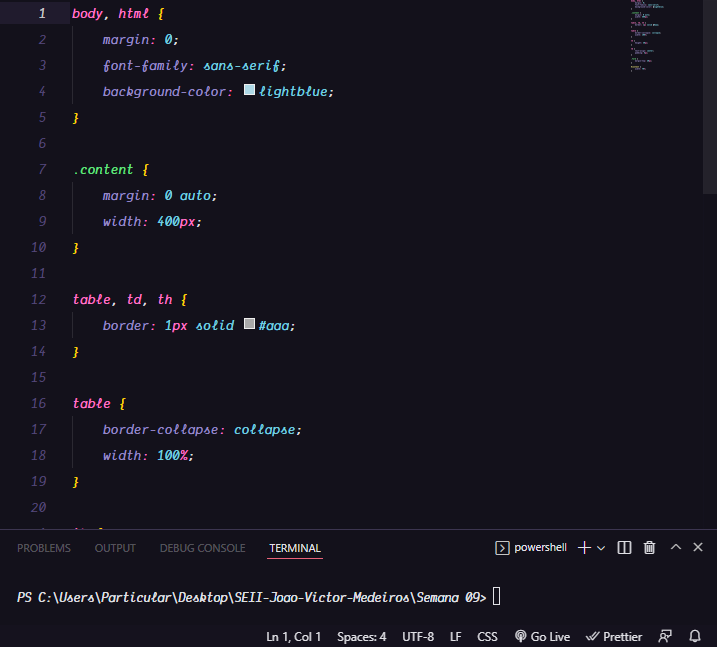


Figura 9: Desenvolvimento do front-end com CSS, código style.css parte 1.

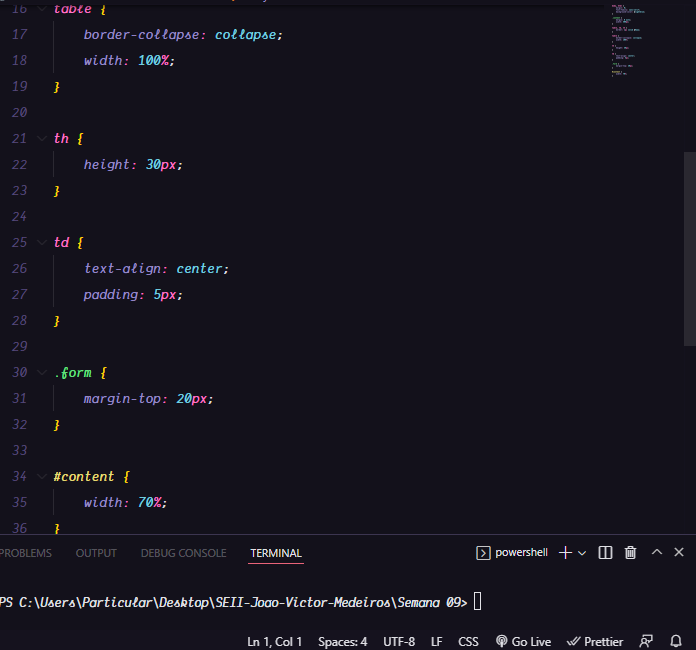


Figura 10: Desenvolvimento do front-end com CSS, código style.css parte 2.

Sendo assim, com tudo criado rodamos a aplicação e obtivemos como resultado a seguinte página HTML onde é possível cadastrar tarefas ou atividades e as remover veja abaixo uma imagem que mostra como ficou o projeto:



Figura 11: Tela final da aplicação.