

Documentação da Macroarquitetura do Projeto "To Do Adventure"

Macroarquitetura:

Visão Geral:

O código implementa uma API RESTful usando Flask para gerenciar tarefas e "Jornadas do Herói".

As tarefas são organizadas em status: "disponíveis", "em andamento" e "concluídas".

As Jornadas são coleções de tarefas que representam um objetivo maior.

Componentes do Sistema:

O código implementa uma API RESTful usando Flask para gerenciar tarefas e "Jornadas do Herói".

- A API usa um arquivo JSON (db.json) para persistência de dados.
- Servidor de Aplicações: Executa a lógica de negócio da aplicação.
- Cache: Armazena dados em memória para otimizar o desempenho.

O principal é o app.component onde fica nosso dashboard. Depois dele a gente tem mais 3 dialogs: O componente dialog que possui o form para criação e edição de novas tarefas, um dialog para a criação e visualização de jornada de herói e outro dialog que serve de template para avisos de confirmação como "deseja salvar?".

Relações entre os Componentes:

- A API recebe requisições dos clientes e as direciona para o servidor de aplicações.
- O servidor de aplicações interage com o banco de dados para atender às requisições.