

Instruções de Setup do Backend (TicketHub)

Este guia explica como configurar e rodar o backend do TicketHub (feito em Python/FastAPI) localmente na sua máquina (Mac) para que você possa desenvolver o frontend em Swift.

O backend depende de três serviços de banco de dados: **MySQL**, **MongoDB** e **Redis**.

Pré-requisito: Homebrew

O Homebrew é o gerenciador de pacotes mais fácil para o macOS. Se você não o tem, instale-o primeiro. Abra seu **Terminal** e rode:

```
/bin/bash -c "$(curl -fsSL  
[https://raw.githubusercontent.com/Homebrew/install/HEAD/install.sh](https://raw.githubusercontent.com/Homebrew/install/HEAD/install.sh))"
```

Passo 1: Instalar os Bancos de Dados (MySQL, Mongo, Redis)

No seu terminal, instale os três serviços usando o Homebrew:

```
# 1. Instalar o MySQL Server  
brew install mysql  
  
# 2. Instalar o MongoDB  
brew install mongodb-community@7.0  
  
# 3. Instalar o Redis  
brew install redis
```

Depois de instalar, você precisa **iniciar os serviços** para que eles fiquem rodando em segundo plano:

```
brew services start mysql  
brew services start mongodb-community@7.0  
brew services start redis
```

Passo 2: Configurar o Banco de Dados MySQL

Agora que o MySQL está rodando, precisamos criar nosso banco de dados e usuário.

1. **Faça login no MySQL como root** (a senha de root geralmente é vazia por padrão no Mac após a instalação com brew):
mysql -u root -p

(Se ele pedir uma senha, apenas pressione Enter).

2. **Execute o Script SQL:** Você deve ter recebido um arquivo tickethub_db.sql. Coloque-o em uma pasta de fácil acesso (ex: sua pasta Desktop). Dentro do MySQL, rode o comando source para executá-lo.

(Lembre-se de trocar /caminho/para/ pelo caminho real do arquivo)

```
mysql> source /caminho/para/tickethub_db.sql;
```

Este comando irá criar o banco tickethub_db, todas as tabelas, funções, e o usuário dev_backend.

3. Digite exit para sair do MySQL.

Passo 3: Configurar o Projeto Python

1. **Clone o Repositório** (ou descompacte o .zip) contendo o código do backend (a pasta app/).
2. Navegue até a pasta raiz do projeto no seu terminal.

3. **Crie um Ambiente Virtual:**

```
python3 -m venv venv
```

4. **Ative o Ambiente Virtual:**

```
source venv/bin/activate
```

(Seu prompt do terminal deve agora mostrar (venv) no início)

5. Instale as Dependências:

Na pasta do projeto, você encontrará o arquivo requirements.txt. Instale tudo com este comando:

```
pip install -r requirements.txt
```

Passo 4: Rodar a API do Backend!

Se tudo deu certo, você está pronto. Com o ambiente virtual (venv) ainda ativado, rode o servidor Uvicorn:

```
uvicorn app.main:app --reload
```

O terminal deve mostrar:

INFO: Uvicorn running on http://127.0.0.1:8000 (Press CTRL+C to quit)

Passo 5: Testar

Abra seu navegador (Chrome ou Safari) e acesse:

`http://localhost:8000/docs`

Você verá a documentação do FastAPI. Agora você pode se registrar (`/register`), fazer login (`/login`) e testar todos os endpoints.

O seu app Swift deve agora fazer requisições para `http://localhost:8000`.

Nota sobre Conexões:

O backend (Python) está configurado em `app/database.py` para se conectar ao MySQL, MongoDB e Redis em localhost. Como você instalou os três serviços localmente (Passo 1), tudo deve funcionar sem nenhuma alteração no código.