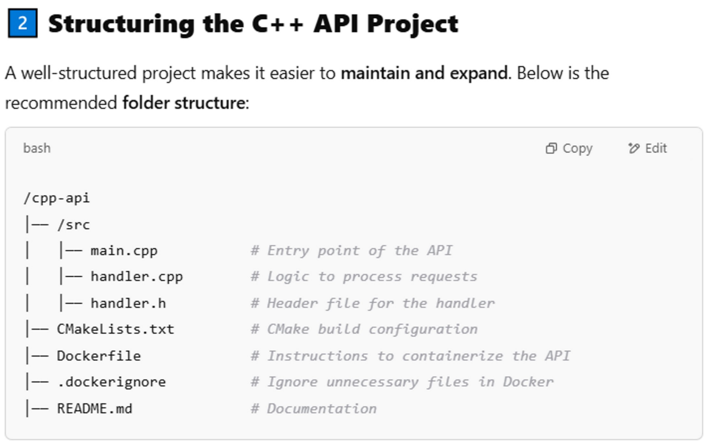
API

terça-feira, 1 de abril de 2025 07:32

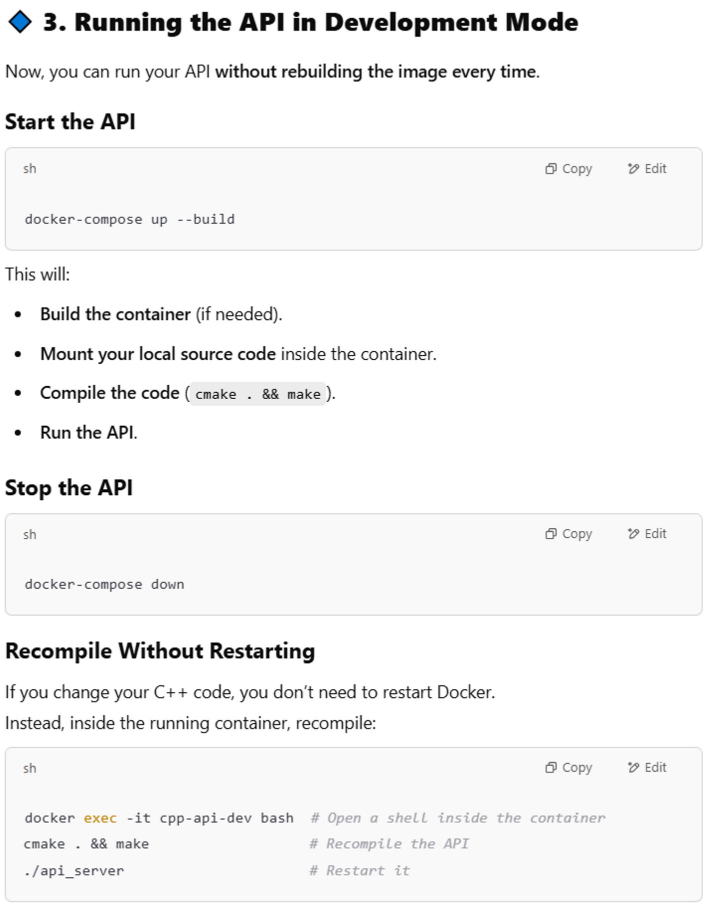
- CppRestSDK is not an option anymore, no support for years and the github page recommend not to use it
- Pistache
  - May lack documentation…. but it's something I can contribute… they are looking for voluntiers
- **Crow**
  - Seems like a good option, has support (github page s active) https://github.com/CrowCpp/Crow
  - Looks like there are some documentation arround it https://crowcpp.org/master/getting_started/setup/linux/
  - GPT gave a solid answer… that is nice



2️⃣ Structuring the C++ API Project

A well-structured project makes it easier to **maintain and expand**. Below is the recommended **folder structure**:

```bash
/cpp-api
|── /src
│   |── main.cpp          # Entry point of the API
│   |── handler.cpp       # Logic to process requests
│   |── handler.h         # Header file for the handler
|── CMakeLists.txt        # CMake build configuration
|── Dockerfile            # Instructions to containerize the API
|── .dockerignore         # Ignore unnecessary files in Docker
|── README.md             # Documentation
```



🔷 3. Running the API in Development Mode

Now, you can run your API **without rebuilding the image every time**.

Start the API

```sh
docker-compose up --build
```

This will:

- **Build the container** (if needed).
- **Mount your local source code** inside the container.
- **Compile the code** (`cmake . && make`).
- **Run the API**.

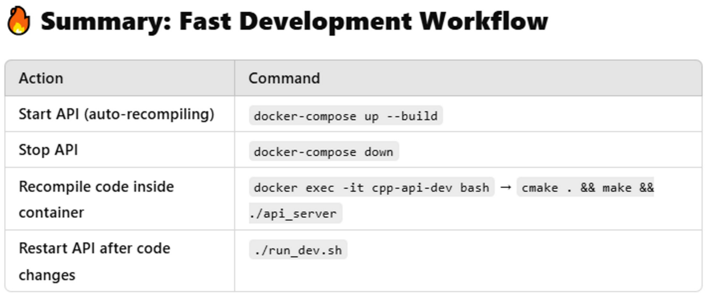
Stop the API

```sh
docker-compose down
```

Recompile Without Restarting

If you change your C++ code, you don't need to restart Docker.
Instead, inside the running container, recompile:

```sh
docker exec -it cpp-api-dev bash  # Open a shell inside the container
cmake . && make                   # Recompile the API
./api_server                      # Restart it
```



🔥 Summary: Fast Development Workflow

Action	Command
Start API (auto-recompiling)	`docker-compose up --build`
Stop API	`docker-compose down`
Recompile code inside container	`docker exec -it cpp-api-dev bash` → `cmake . && make && ./api_server`
Restart API after code changes	`./run_dev.sh`

1️⃣ Build & Start Container

docker-compose up --build

2️⃣ Manually Rebuild (if needed)

```
docker exec -it cpp-api-dev bash
cd build && cmake .. && make -j$(nproc)
./api_server  # Run manually
```

3️⃣ Stop Container

docker-compose down