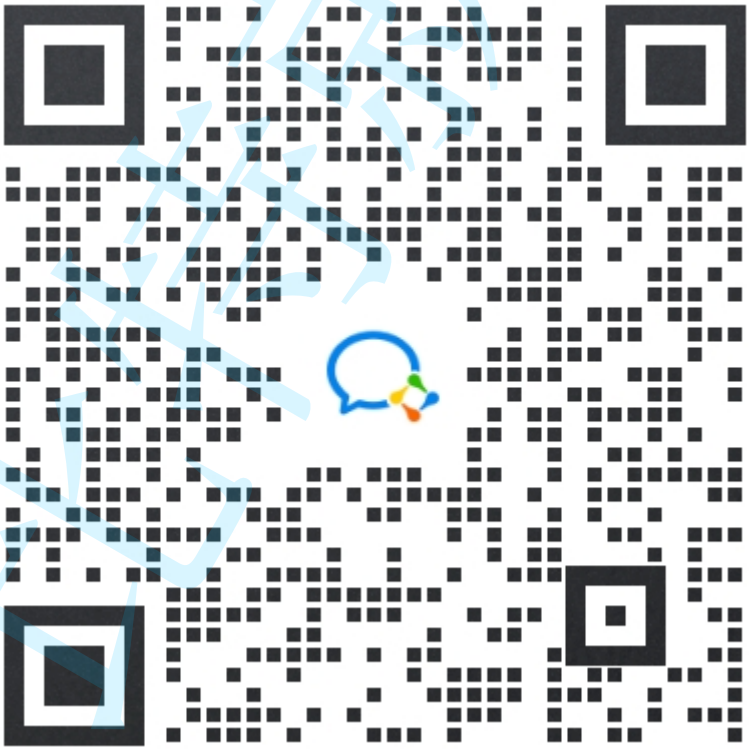
cpp-httplib 安装与使用

版权说明

对比特项目感兴趣，可以联系这个微信。



代码 & 板书链接

https://gitee.com/bitedu-tech/cpp-chatsystem

介绍

C++ HTTP 库（cpp-httplib）是一个轻量级的 C++ HTTP 客户端/服务器库，它提供了简单的 API 来创建 HTTP 服务器和客户端，支持同步和异步操作。以下是一些关于 cpp-httplib 的主要特点：

1. **轻量级**：cpp-httplib 的设计目标是简单和轻量，只有一个头文件包含即可，不依赖于任何外部库。
2. **跨平台**：它支持多种操作系统，包括 Windows、Linux 和 macOS。
3. **同步和异步操作**：库提供了同步和异步两种操作方式，允许开发者根据需要选择。
4. **支持 HTTP/1.1**：它实现了 HTTP/1.1 协议，包括持久连接和管道化。
5. **Multipart form-data**：支持发送和接收 multipart/form-data 类型的请求，这对于文件上传非常有用。
6. **SSL/TLS 支持**：通过使用 OpenSSL 或 mbedTLS 库，cpp-httplib 支持 HTTPS 和 WSS。
7. **简单易用**：API 设计简洁，易于学习和使用。
8. **性能**：尽管是轻量级库，但性能表现良好，适合多种应用场景。
9. **社区活跃**：cpp-httplib 有一个活跃的社区，不断有新的功能和改进被加入。

安装

```C++
dev@dev-host:~/workspace$ git clone
https://github.com/yhirose/cpp-httplib.git
```

类与接口

```C++
namespace httplib {
struct Request {
  std::string method;
  std::string path;
```

```
  Headers headers;
  std::string body;
  Params params;
}
struct Response {
  std::string version;
  int status = -1;
  std::string reason;
  Headers headers;
  std::string body;
  void set_content(const std::string &s,
      const std::string &content_type);
  void set_header(const std::string &key,
      const std::string &val);
}
class Server {
    using Handler = std::function<void(const Request &, Response
&)>;
    Server &Get(const std::string &pattern, Handler handler);
    Server &Post(const std::string &pattern, Handler handler);
    Server &Put(const std::string &pattern, Handler handler);
    Server &Delete(const std::string &pattern, Handler handler);
    bool listen(const std::string &host, int port);
}
class Client {
    explicit Client(const std::string &host, int port);
    Result Get(const std::string &path, const Headers &headers);
    Result Post(const std::string &path, const std::string &body,
                const std::string &content_type);
    Result Put(const std::string &path, const std::string &body,
                const std::string &content_type);
    Result Delete(const std::string &path, const std::string
&body,
                const std::string &content_type);
}
}
```

使用

```C++
#include "cpp-httplib/httplib.h"

class HelloServer {
```

```
    public:
        HelloServer(int port):_port(port) {
            _server.Get("/hi", std::bind(
                &HelloServer::HelloWorld, this,
                std::placeholders::_1,
                std::placeholders::_2));
        }
        void run() {
            _server.listen("0.0.0.0", _port);
        }
    public:
        void HelloWorld(const httplib::Request &req,
            httplib::Response &rsp) {
            std::string body = "<h1>HelloWorld</h1>";
            rsp.set_content(body, "text/html");
            rsp.status = 200;
        }
    private:
        int _port;
        httplib::Server _server;
};

int main()
{
    HelloServer server(8080);
    server.run();
    return 0;
}
```

```
C++
main : main.cc
    g++ -std=c++17 $^ -o $@ -pthread
```