```
                              server
                                 │
                                 ▼
  std::string save_new_page(const std::string& url)
                                 │
                                 ▼
                           http_client                          file_manager
                                                        source new_source;
  - int socket_fd;                                       std::string file_name;
  - http_request request;                        ◁
```
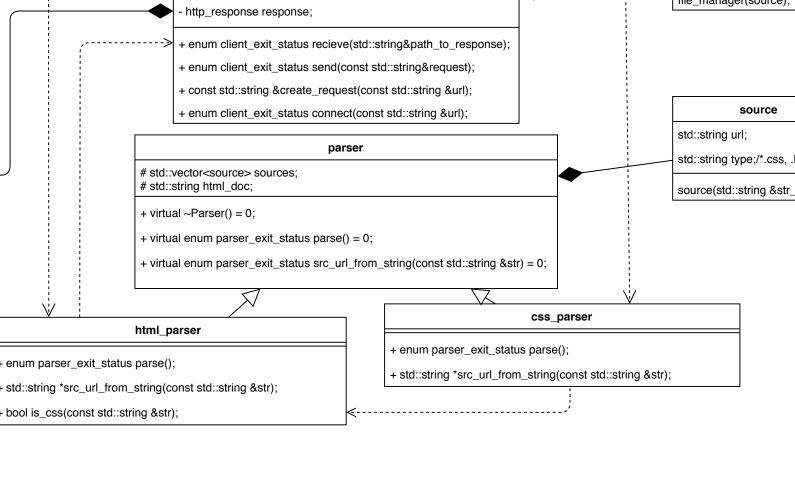
http_request

std::string query;

std::string host;

+ http_request(std::string& url)

http_response

std::string query;

std::string html_body;

+ http_response(std::string& response);

- http_response response;

+ enum client_exit_status recieve(std::string&path_to_response);

+ enum client_exit_status send(const std::string&request);

+ const std::string &create_request(const std::string &url);

+ enum client_exit_status connect(const std::string &url);

---

parser

std::vector<source> sources;
std::string html_doc;

+ virtual ~Parser() = 0;

+ virtual enum parser_exit_status parse() = 0;

+ virtual enum parser_exit_status src_url_from_string(const std::string &str) = 0;

---

source

std::string url;

std::string type;/*.css, .l

source(std::string &str_

---

file_manager(source);

---

html_parser

+ enum parser_exit_status parse();

+ std::string *src_url_from_string(const std::string &str);

+ bool is_css(const std::string &str);

---

css_parser

+ enum parser_exit_status parse();

+ std::string *src_url_from_string(const std::string &str);

html, .js*/
from_htm