Class HttpRouter {

Static <name, callback> routes[]

Static <name, callbacn> filters[];

Static httprouter &instance {

}

Static add\_router/add\_filter

Static void routing(int client\_id, base\_url, req\_type) – calls filter and route

};

Class HttpController {

Static init\_path\_routing()

{ HttpRouter::instance.add\_router}

}

Class HttpFilter {

Static\_path\_routing

{ httprouter::instance\_add\_filter }

}

Class httpserver {

Static httpserv& instance() {

}

When it accepts a new connection it processes it and then calls:

HttpRouter::routing(client\_id, url, type)

}

Request response structure

Class Response {

Void add\_header\_raw();

Void add\_header(enum Header, str value);

// Basic headers

Void set\_content\_type();

Void set\_auth\_bearer();

Void set\_cookie();

Private:

String body;

String http\_version;

}