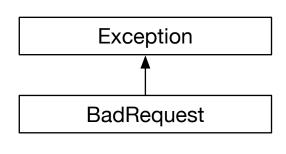
Web Server Project Class Diagram Proposal (v1.2)



- uri : String - body : ? verb : String
- httpVersion : String - headers : Dictionary
- + parse(): void + accessors

ResponseFactory

+ getResponse(request : Request, resource : Resource) : Response

Response

- + code : integer
- + reasonPhrase : String + resource : Resource
- + Response(resource : Resource)
- + send(out : OutputStream)

Request

- + Request(test : String)
- + Request(client : Stream)

MimeTypes

- types : Dictionary
- + MimeTypes(fileName : String)
- + load(): void
- + lookup(extension : String) : String

HttpdConf

- aliases : Dictionary
- scriptAliases : Dictionary
- etc
- + HttpdConf(fileName :
- String)
- + load(): void

Htaccess

- userFile: Htpassword
- authType: String
- authName: String
- require : String
- + load(): void
- + isAuthorized(username : String, password : String) : boolean

Htpassword

- users : Dictionary

ConfigurationReader

+ ConfigurationReader(fileName: String)

+ hasMoreLines(): boolean

+ nextLine(): String

+ load(): void

- file : File

- + load(): void
- + isAuthorized(username : String,
- password : String) : boolean

- configuration: HttpdConf

- accessFiles: Dictionary
- + start() : void

Server

- mimeTypes : MimeTypes socket: ServerSocket

Logger

- file : File
- + Logger(fileName: String)
- + write(request : Request, response :

Response)

Resource

- + Resource(uri : String, config :
- HttpdConf)
- + absolutePath(): String + isScript(): boolean + isProtected(): boolean

Worker

Thread

- client : Socket
- mimes : MimeTypes - config: HttpdConf
- + Worker(socket : Socket, config : HttpdConf, mimes : MimeTypes)
- + run()