目录

[1 实验目的和要求 1](#_Toc423006326)

[2 实验环境 1](#_Toc423006327)

[3 实验内容 1](#_Toc423006328)

[3.1 此Web Server的主要特性： 1](#_Toc423006329)

[3.2 构建目录 1](#_Toc423006330)

[3.2.1 目录结构 1](#_Toc423006331)

[3.2.2 源码结构 2](#_Toc423006332)

[3.2.3 其他文件夹 3](#_Toc423006333)

[3.3 处理流程 3](#_Toc423006334)

[4 测试结果 4](#_Toc423006335)

[4.1 启动、运行及关闭 4](#_Toc423006336)

[4.1.1 不带GUI界面的启动： 5](#_Toc423006337)

[4.1.2 带GUI界面的启动： 5](#_Toc423006338)

[4.2 服务器工作测试（以Chrome为例） 7](#_Toc423006339)

[4.3 日志记录： 11](#_Toc423006340)

[5 实验总结 12](#_Toc423006341)

[5.1 过程 12](#_Toc423006342)

[5.2 细节 13](#_Toc423006343)

[5.3 引用： 13](#_Toc423006344)

1. 开发环境

（1）语言：Java SE 1.7

（2）开发工具：Eclipse

（3）运行系统：装有java运行环境（JRE）的Windows/Linux系统

1. 项目内容
   1. 此Web Server的主要特性：

（1）实现了200（成功）、2S6（部分内容）、307（临时重定向）、304（未修改）、400（请求无效）、404（文件不存在）、500（服务器内部错误）状态码的处理与响应。

（2）支持HTML（包含图片、音频、视频、Flash等）、CSS、JavaScript等资源的请求。

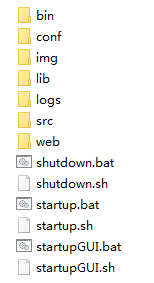
（3）运用多线程并发处理请求响应，以使多源同时访问不阻塞。

（4）附带GUI界面与脚本开关两种方式，支持Windows/Linux系统下运行。

（5）用户可以设置端口号、编码、默认项目、是否开启304等设置。

（6）如果请求根目录，根据是否配置默认项目，合理地重定向页面。

* 1. 构建目录
     1. 目录结构



（1）src为源码java文件

（2）bin为编译后的class文件

（3）conf为xml配置文件

（4）lib为引用的jar包

（5）logs为日志文件

（6）img为GUI所用的图片文件

（7）web为项目所在根目录，内含两个测试项目。

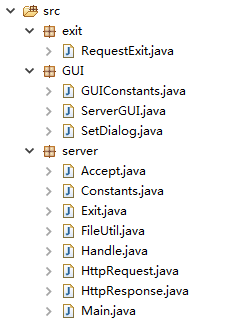
（8）startup.bat/startup.sh为Win/Linux下的无界面的启动脚本

shutdown.bat/shutdown.sh为Win/Linux下的无界面的关闭脚本

startupGUI.bat/ startupGUI.sh为Win/Linux下的带GUI界面的启动脚本

发布Web项目只需把项目文件夹放到web文件夹下即可。

* + 1. 源码结构



Server包主要是不带GUI界面的Web Server核心类。包含：

（1）Main 入口函数所在类。启动、关闭服务器的主类。

（2）HttpRequest和HttpResponse是对请求和响应的封装类

（3）FileUtil为文件操作公共方法类，这里主要是遍历其下所有子文件及子文件夹，主要用于服务器启动时更新文件最后修改时间。

（4）Constants为常量配置类

（5）Exit类为服务器监听关闭接口等待关闭命令的类

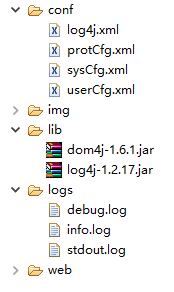
（6）Accept为服务器无限循环监听请求的线程类

（7）Handle为解析请求，生成响应的核心任务逻辑线程类

Exit包是单独的java小程序，用于在无GUI界面的情况下顺利关闭Web Server。

GUI包是Web Server的GUI界面外壳。

* + 1. 其他文件夹



conf下：

（1）log4j.xml为日志记录器配置文件

（2）protCfg.xml为协议配置，主要为读入状态码表及文件类型表的常量。

（3）sysCfg.xml为系统配置，主要为程序运行一般不更改的系统常量。

（4）userCfg.xml为用户配置，用户看更改如端口、默认项目、编码等等。默认为80 端口为web访问的规定端口，用户可更改为其他端口作为调试。

logs下：

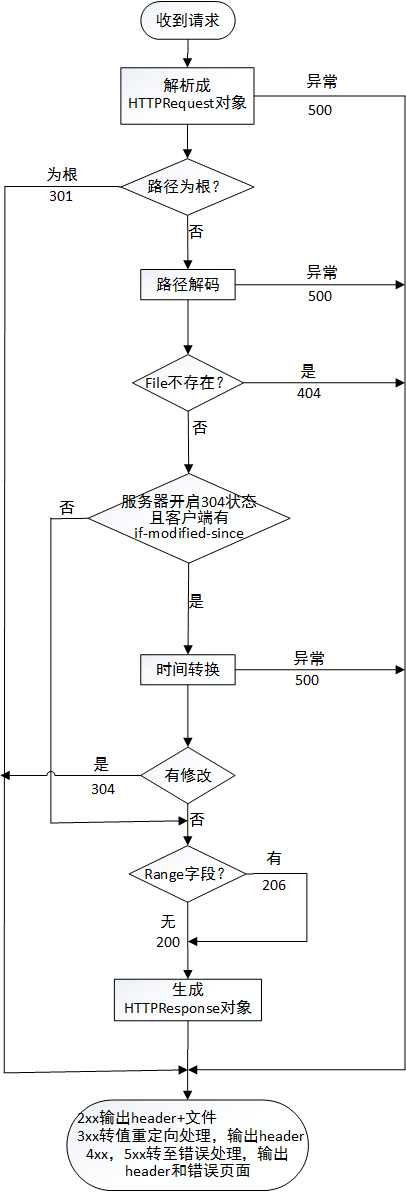
（1）debug.log主要记录请求与响应的头部字段，作为调试参考，文件大小限定500k。

（2）info.log主要记录服务器加载配置、运行、停止等系列信息，文件大小限定500k。

（3）stdout.log为系统输入输出及异常、错误输入输出的记录，文件大小限定500k。

* 1. 处理流程

启动、关闭服务器略。



1. 测试结果
   1. 启动、运行及关闭
      1. 不带GUI界面的启动：

（1）Windows：

运行脚本startup.bat启动Web Server，运行脚本shutdown.bat关闭Web Server。

（2）Linux：

运行脚本startup.sh启动Web Server，运行脚本shutdown.sh关闭Web Server。

* + 1. 带GUI界面的启动：

Win下运行脚本startupGUI.bat，Linux下运行startupGUI.sh启动Web Server（以Windows8.1为例）：

整个界面主要分成两个部分：菜单区和信息区。

（1）“启动”和“停止”按钮启动/停止服务器，当服务器未开启时，“停止”按钮不可用。当服务器启动时，“启动”按钮不可用。如图4-1-1、4-1-2所示。



图4-1-1

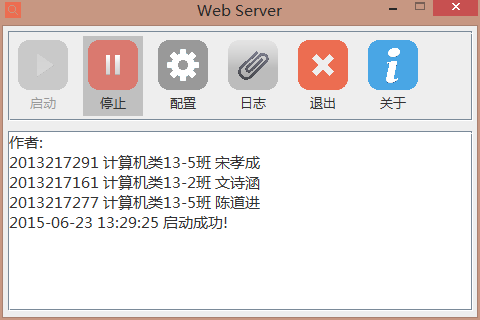


图4-1-2

（2）“配置”按钮，打开配置对话框，在配置对话框中可设置端口、字符集、默认项目目录以及是否启用304（未修改）状态响应。此外，还可以进入配置文件进行手动配置。如图4-1-3所示。

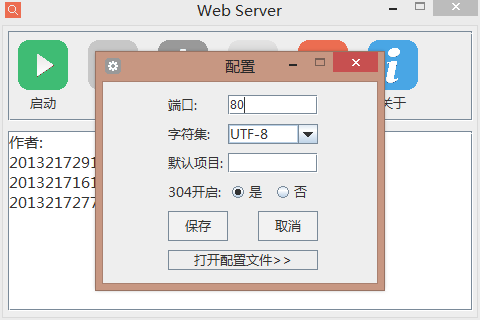


图4-1-3

（3）“日志”按钮，可打开日志所在的文件夹。

（4）“退出”按钮，退出程序。

（5）托盘区菜单。本界面禁止调节界面大小，最小化和关闭按钮均将程序最小化到托盘。托盘区浮出菜单有：打开、启动、停止和退出，功能依次为：打开主面板、启动服务器、停止服务器和退出程序，如图4-1-4所示。



图4-1-4

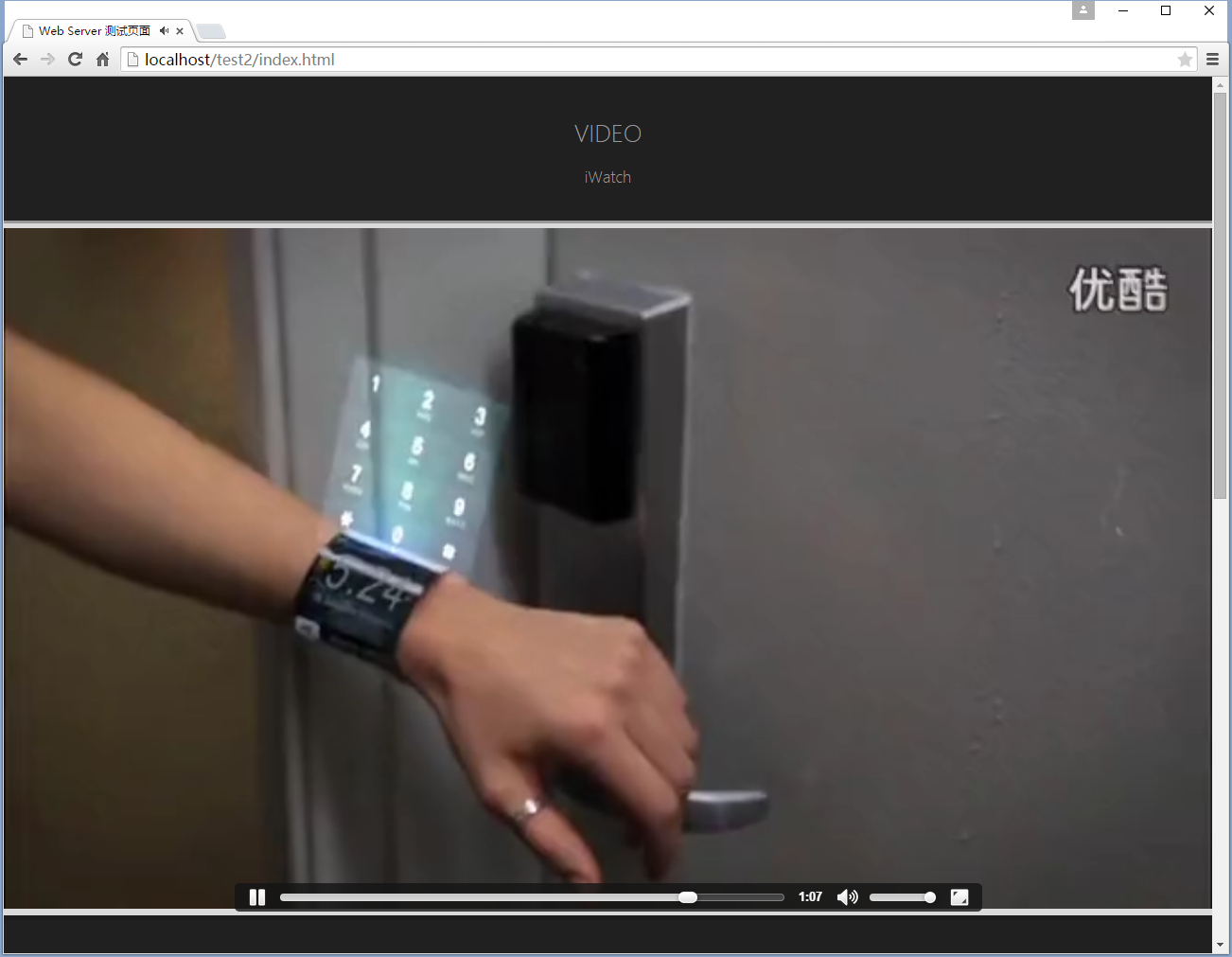
* 1. 服务器工作测试（以Chrome为例）

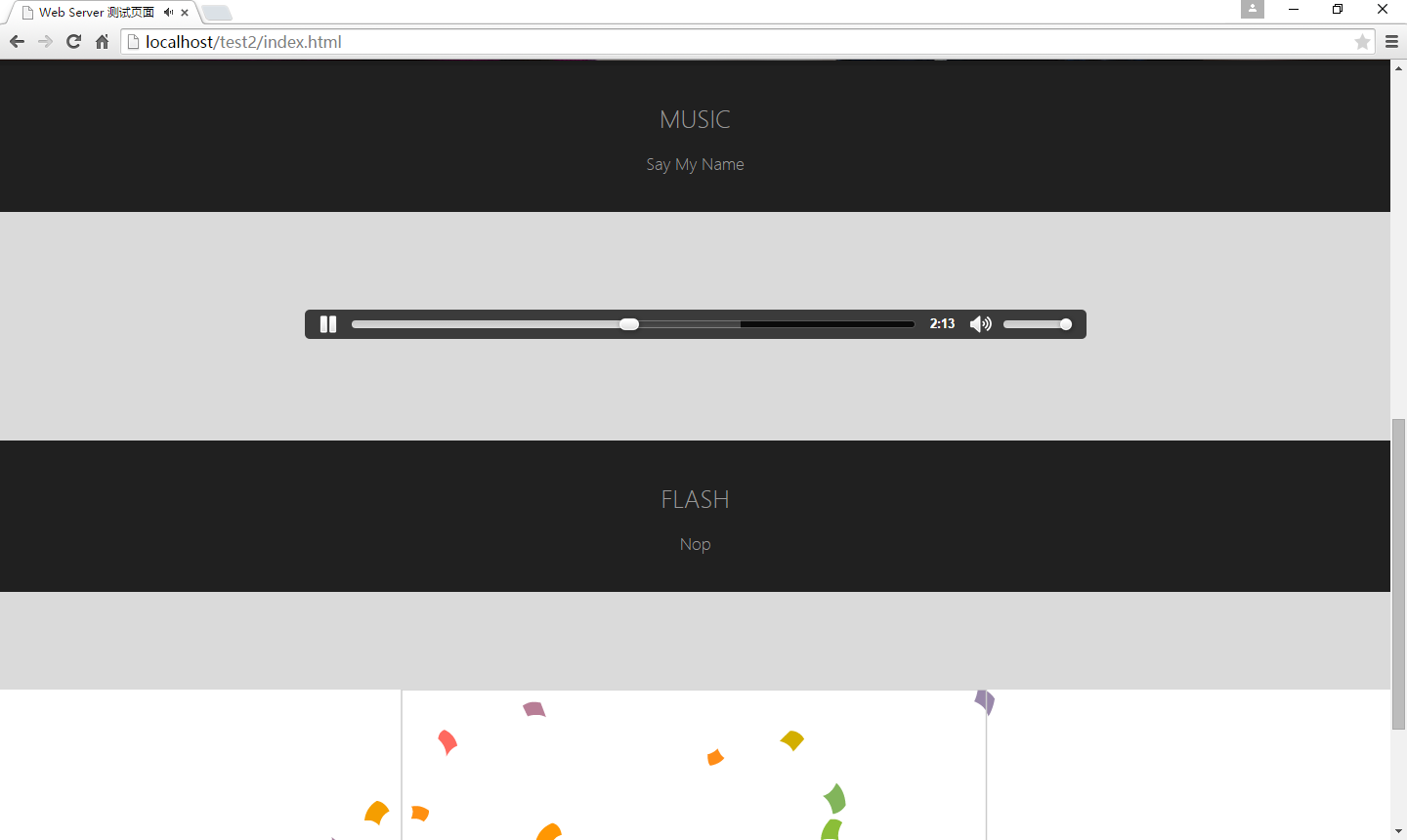
发布Web项目只需把项目文件夹放到web文件夹下即可，”[主机名]/”对应web目录，作品里web文件夹下自带两个测试项目test1和test2。

（1）测试页面1（HTML（图片）、CSS（图片）、JavaScript）：



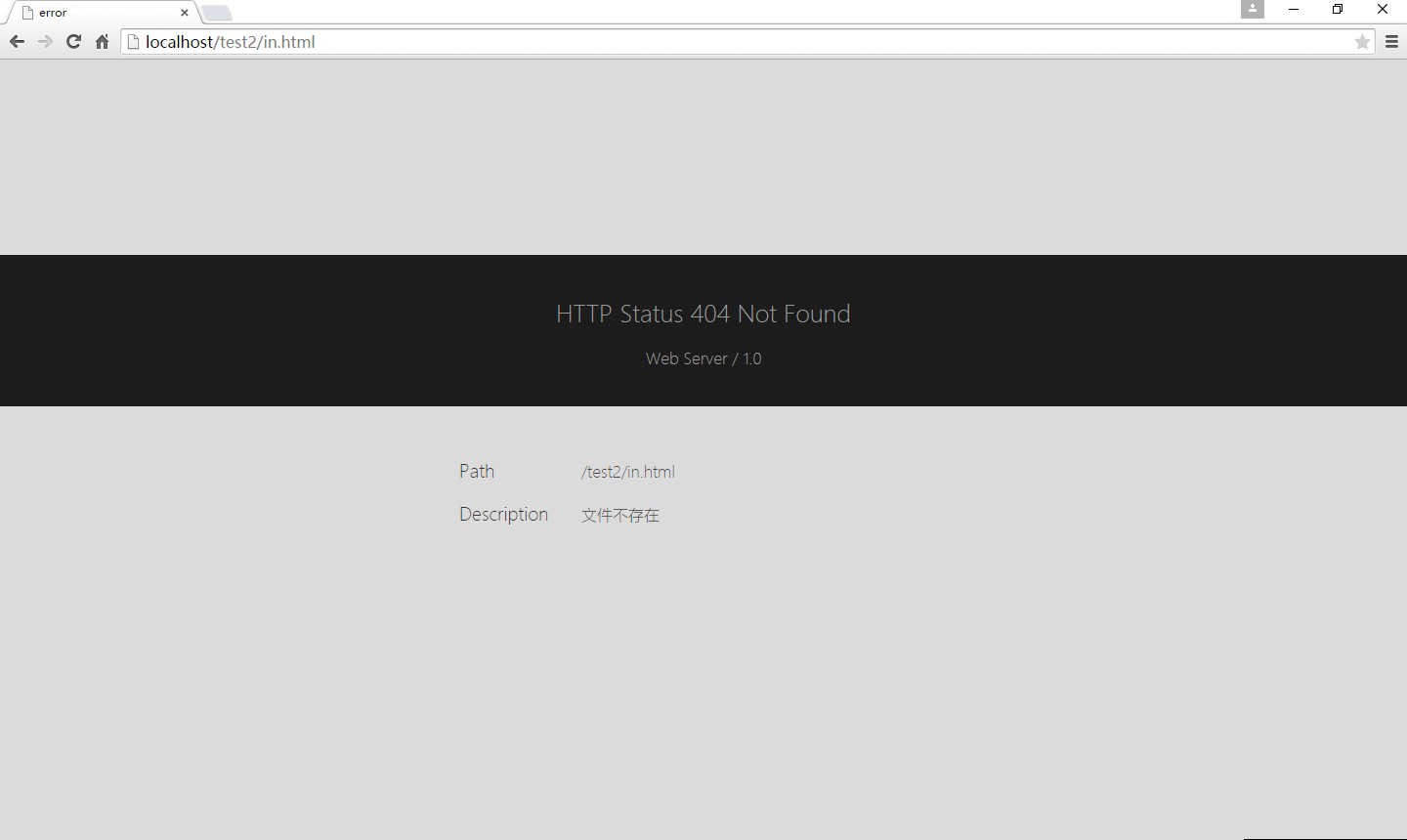
（2）测试页面2（HTML（图片、音频、视频、Flash）、CSS）





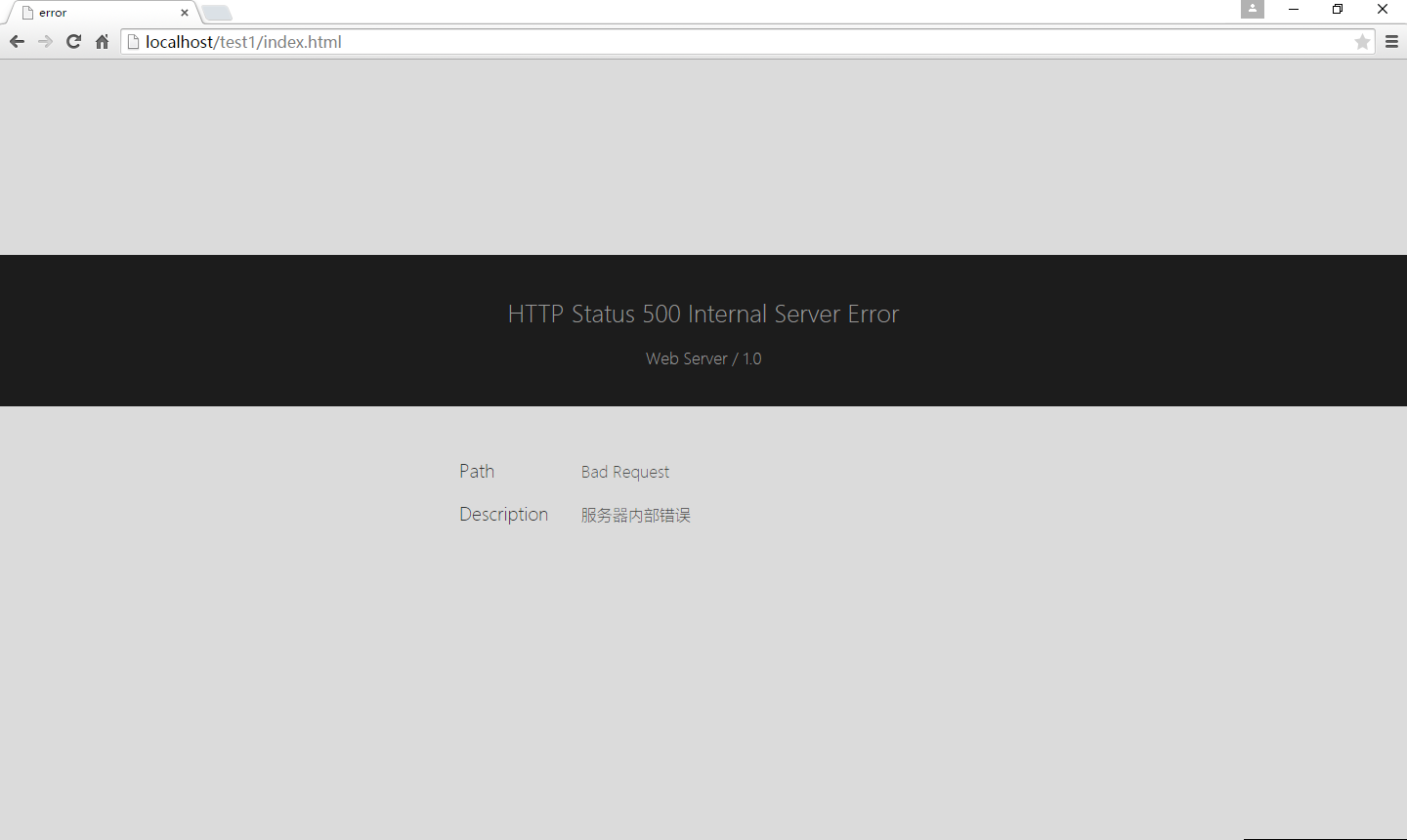
（3）404测试

我们故意访问不存在的文件，造成找不到文件错误，测试404 页面



（4）500测试

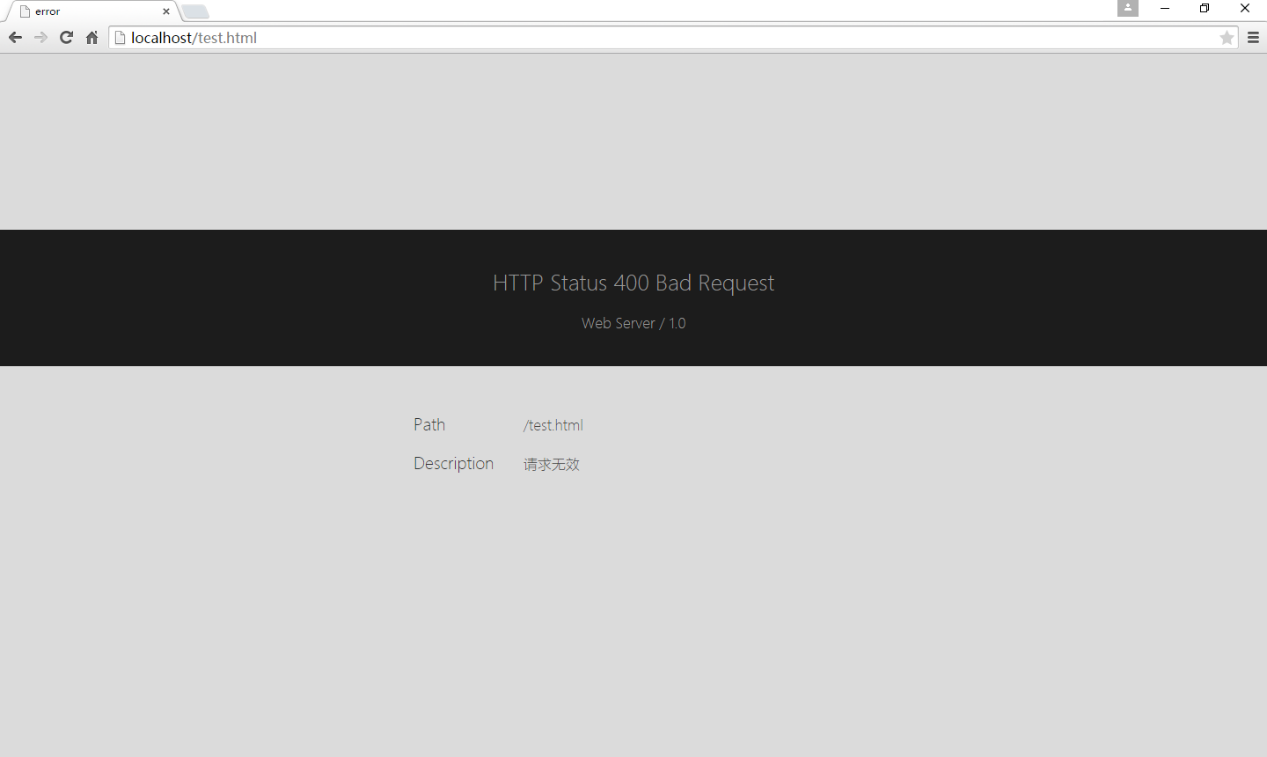
我们故意设置编码为不存在的sxc，造成服务器错误，测试500页面



改回UTF-8即又正常访问。

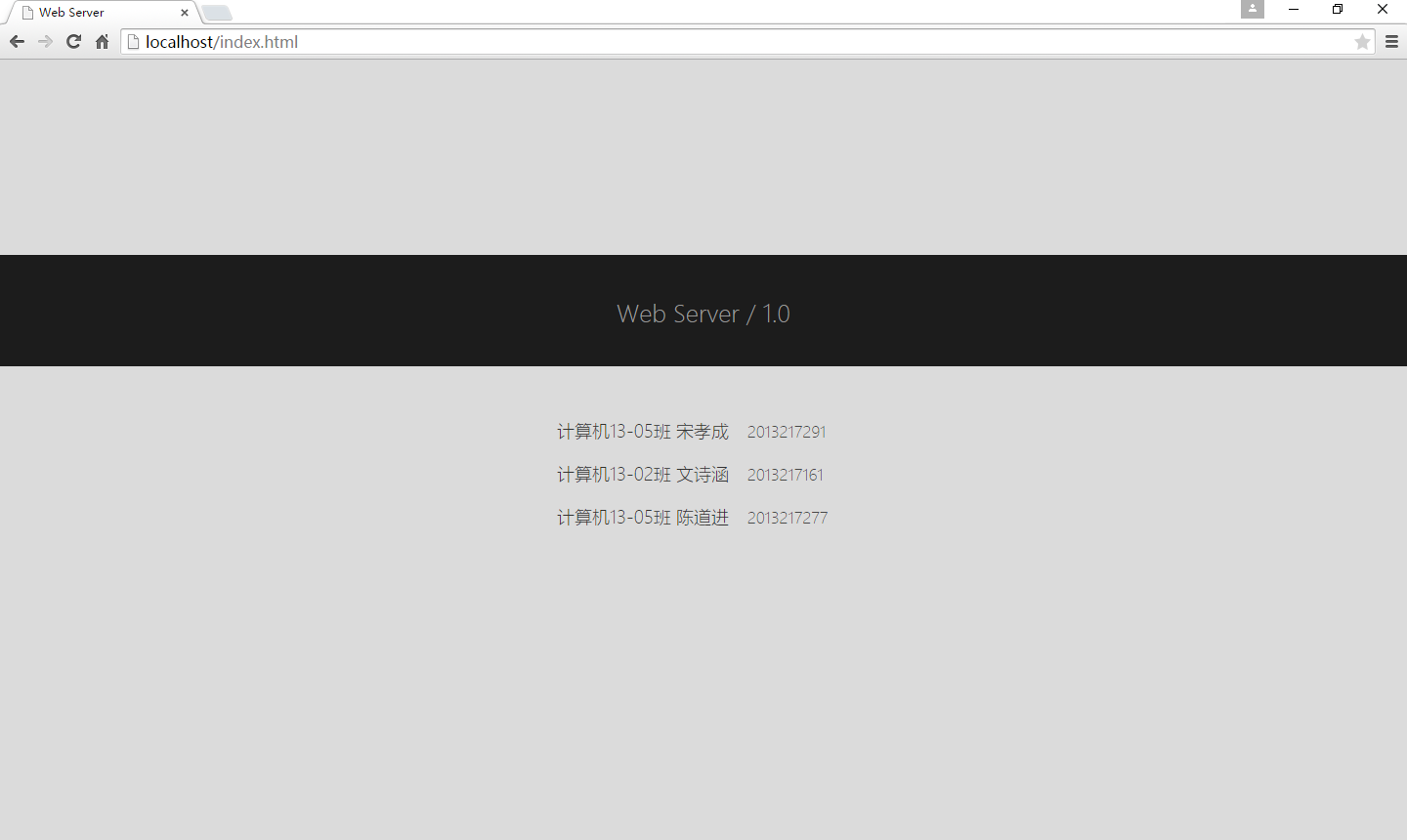
（5）400测试

我们故意访问[主机名]/[文件名]（正常应该是[主机名]/[项目名]/[文件相对路径]），造成请求无效错误，测试400页面

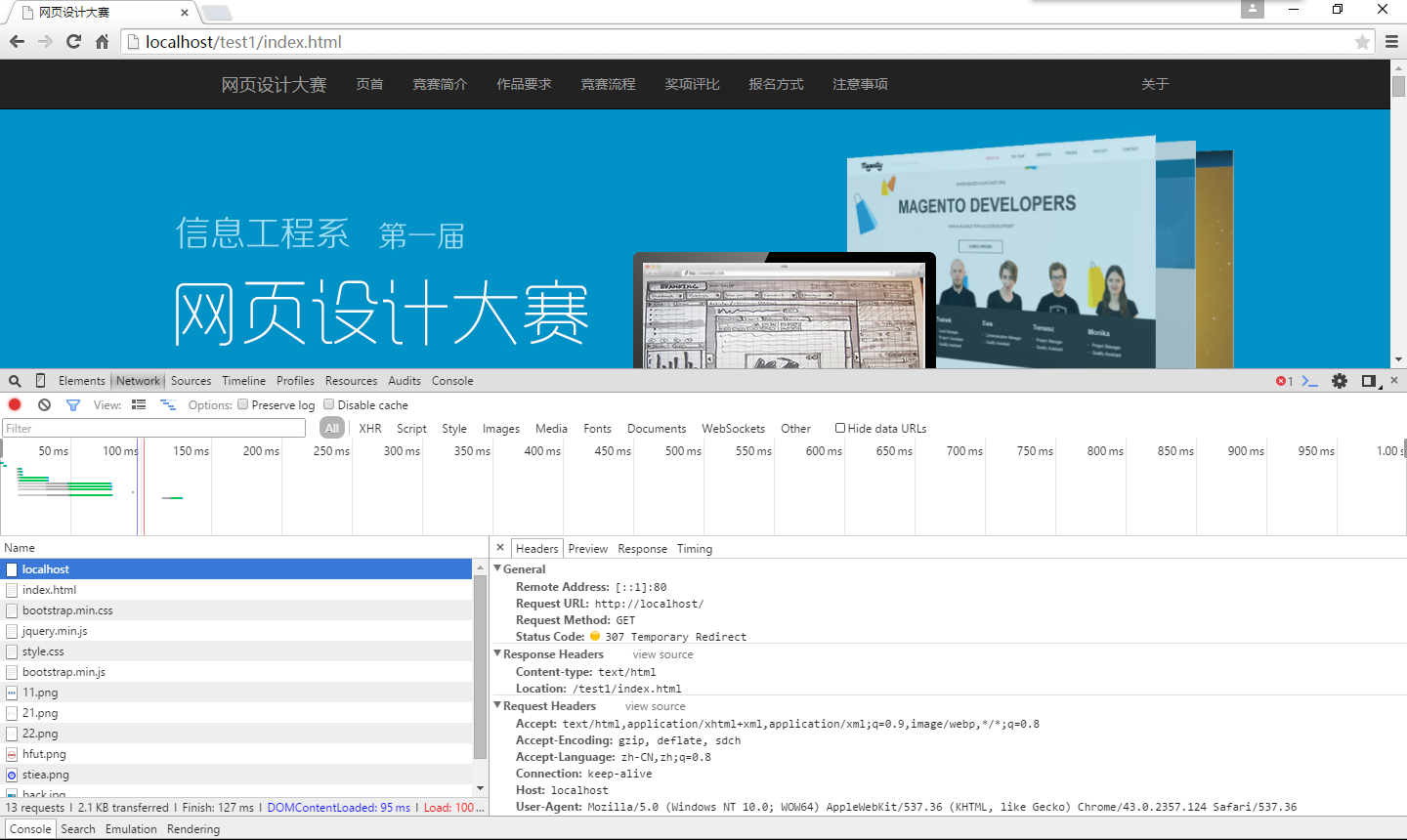


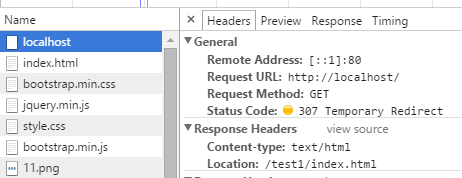
（6）307测试

未设置默认项目时，访问服务器根目录：[主机名] 回车 或 [主机名]/ 回车，重定向至默认的Web Server首页



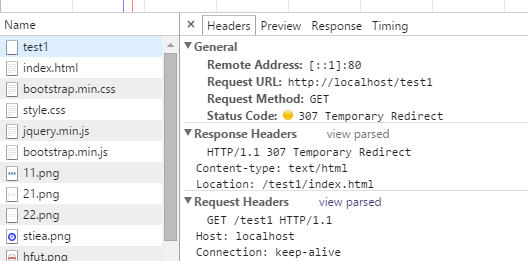
设置了默认项目（以test1为例）时，访问服务器根目录，[主机名] 回车 或 [主机名/] 回车，重定向至默认项目的Index.html首页。





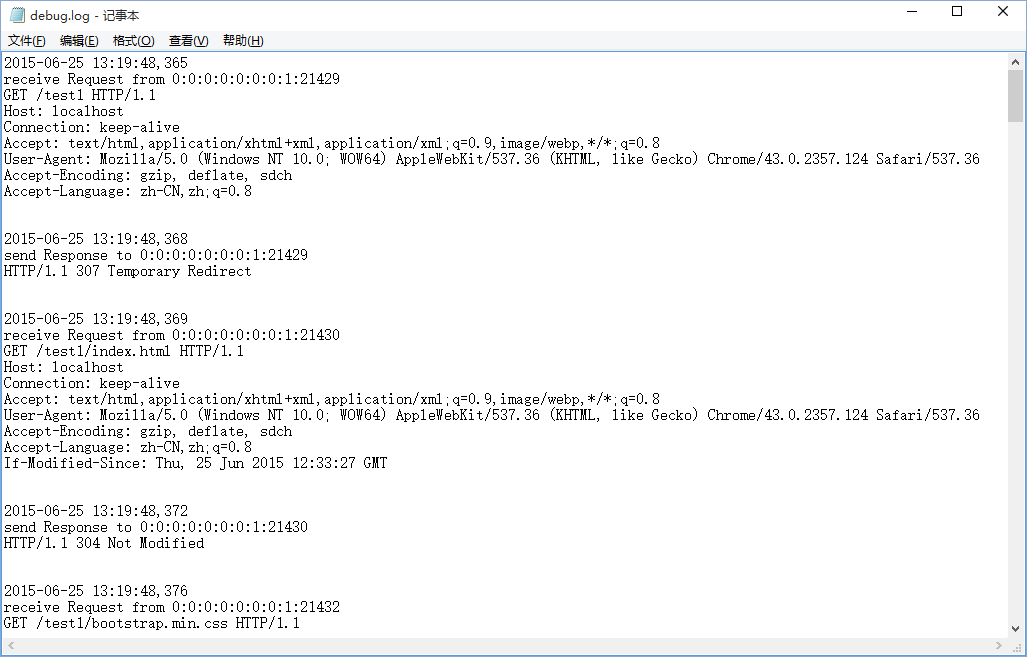
访问项目根目录：

以test1为例，显示页面同上，浏览器开发者工具如下：

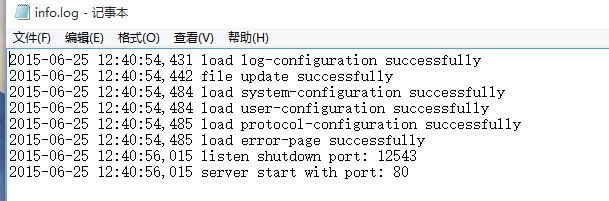


* 1. 日志记录：

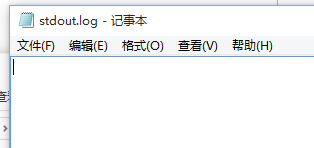
运行以上测试后，查看日志文件如下：



debug.log



info.log



stdout.log

1. 总结
   1. 过程

（1）最简单的Web Server 核心功能实现：服务器监听端口，有连接请求，建立Socket连接，从socket输入流按协议读入请求，根据请求路径将响应头和相应文件按照http协议写入socket的输出流，即完成核心功能。

（2）在核心功能基础上，加入条件判断、文件检查、未修改返回、重定向等等多种常用状态码处理充实Web Server功能。

（3）在此基础上，改为多线程并行处理任务，对于提高服务器效率，减少浏览器等待时间等等都有很大的作用。

（4）改为配置文件配置常量，加入日志记录功能，附加GUI界面，增强用户体验，方便操作运行，使之成为一款能用的作品。

（5）距离完整的Web Server还有很远的距离，比如，还有很多不常见的状态处理没有实现，服务器端缓存等其他提高服务器效率的措施没有实现等。

* 1. 细节

（1）Http协议规定头部和实体之间有一行空行，作为头部和实体分隔的标记。

（2）对于Chrome等浏览器，音视频文件如果返回状态为200 ，则无法进行进度条操作，需要返回206状态，并附上正确的Content-Range字段则可以进行进度条拖动点击等操作。

（3）ServerSocket的accept()是阻塞性的方法，且服务器需要无限循环等待接收请求，故在无界面情况下关闭服务器采用单独开辟一个端口监听停止命令的方式。

（4）文件流传输部分，被访问文件大小不一，差别太大，尤其是对于音视频文件的请求，如果一次读取太大，占用内存资源太多，数组甚至要开到几十上百兆或更多，但是对于大部分html、CSS等文件却只有K单位，故采用固定大小数组，分段循环读取文件，放入输出流，输出流缓冲区满会自动将数据发送出。

* 1. 引用

（1）日志记录器使用log4j-1.2.17.jar by Apache

（2）Xml解析器使用dom4j-1.6.1.jar by Apache

（3）GUI界面所有图标均是从<http://www.easyicon.net/>下载后进行一定的处理后得到的。

2013217161 计算机13-2班 文诗涵

2013217277 计算机13-5班 陈道进

2013217291 计算机13-5班 宋孝成

2015年6月25日