



[Document type]

# Test Cases

*[Subtitle]*



Author:  
Supervisor: [Supervisor]  
Semester: HT20  
Course code: 2DV610



## Table of contents

<b>1</b>	<b>Requirements .....</b>	<b>3</b>
1.1	<i>Req1. The web server should be responsive under high load .....</i>	<i>3</i>
1.2	<i>Req2. The web server must follow minimum requirements for HTTP</i>	
1.1	<i>3</i>	
1.3	<i>Req3. The web server must work on Linus, Mac, Windows* .....</i>	<i>3</i>
1.4	<i>Req4. The source code should be released under GPL-2.0 .....</i>	<i>3</i>
1.5	<i>Req5. The access log should be viewable from a text editor.....</i>	<i>3</i>
<b>2</b>	<b>Use Cases .....</b>	<b>4</b>
2.1	<i>Use Case 1: Start Server .....</i>	<i>4</i>
2.2	<i>Use Case 2: Stop Server .....</i>	<i>5</i>
2.3	<i>Use Case 3: Request Shared Resource.....</i>	<i>6</i>
<b>3</b>	<b>Manual Test Cases .....</b>	<b>7</b>
3.1	<i>Test Case 1: Start Server Successfully .....</i>	<i>7</i>
3.2	<i>Test Case 1: Start Server with Taken Socket Port Number.....</i>	<i>9</i>
3.3	<i>Test Case 2: Stop Server Successfully .....</i>	<i>10</i>
3.4	<i>Test Case 3: Request Shared Resource Not Found.....</i>	<i>12</i>



## 1 Requirements

- 1.1 Req1. The web server should be responsive under high load
- 1.2 Req2. The web server must follow minimum requirements for HTTP 1.1
- 1.3 Req3. The web server must work on Linus, Mac, Windows\*
- 1.4 Req4. The source code should be released under GPL-2.0
- 1.5 Req5. The access log should be viewable from a text editor



## 2 Use Cases

### 2.1 Use Case 1: Start Server

**Primary Actor:** Administrator

**Postcondition:**

- A web server has been started.
- A note in the access log was written, that the server was started.

**Main Scenario:**

1. Starts when an administrator wants to start the server.
2. System asks for socket port number and shared resource container.
3. The administrator provides a socket port number and a shared resource container.
4. The system starts a web server on the given port and presents that the server was started and write a note in the access log.

**Alternative Scenario:**

2a. The web server could not be started due to the socket was taken.

1. The system presents an error message: "Socket XX was taken" (XX is the socket number, Example "80").
2. Exit Use Case.

2b. The web server could not be started due to restriction on the shared resource container.

1. The system presents an error message: "No access to folder XX" (XX is the shared resource container provided, Example "\var\www").
2. Exit Use Case.

2c. The access log could not be written to.

1. The system presents an error message. "Cannot write to server log file log.txt".
2. Exit Use Case.



## 2.2 Use Case 2: Stop Server

**Primary Actor:** Administrator

**Precondition:**

- A web server has been started.

**Postcondition:**

- A note in the access log was written, that the server was stopped.

**Main Scenario:**

1. Starts when a user wants to stop the server.
2. The system stops the web server and presents that the webserver has been stopped.



## 2.3 Use Case 3: Request Shared Resource

**Primary Actor:** Browser

**Precondition:** A web server has been started.

**Postcondition:**

- A note in the access log was written, that access happened with the request information and the result of the request.

**Technical Note:**

- Browser and System communicate using HTTP 1.1.
- Error messages are part of the HTTP 1.1 protocol.
  - 200 OK
  - 400 Bad Request
  - 403 Forbidden
  - 404 Not Found

**Main Scenario:**

1. Starts when a browser wants to access a shared resource.
2. The system delivers the shared resource to the browser and a success message is written to the access log.

**Alternative Scenario:**

- 2a. The shared resource cannot be found.
  1. The system presents that the resource cannot be found.
  2. Exit Use Case.
- 2b. The shared resource is outside the shared resource container.
  1. The system presents that the resource is forbidden.
  2. Exit Use Case.
- 2c. The resource request is invalid or malformed.
  1. The system presents that the request cannot be handled.
  2. Exit Use Case.
- 2d. The server encountered an error when trying to process the request.
  1. The system presents that it has an internal error.
  2. Exit Use Case.



### 3 Manual Test Cases

#### 3.1 Test Case 1: Start Server Successfully

**Use Case:** UC 1. Start Server

**Requirement:** Req3

**Scenario:** Starts when an administrator wants to start the server. The system starts a web server on the given port and presents that the server was started and write a note in the access log.

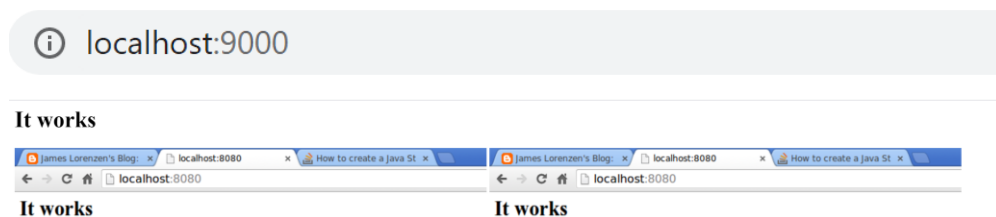
**Precondition:** The server is not deployed yet

**Test Steps:**

1. Set Project JDK to java “version 1.7” and Project language level to “7 – Diamonds, ARM, multi-catch etc”.
2. Edit Configuration for the class “HTTPServerConsole”, and set Program Arguments as “9000  
C:/Github/Repository/2DV610/MyWebServer-master/bin/se/lnu/http/resources/inner/”.
3. In terminal enter “ java se.lnu.http.HTTPServerConsole 9000  
~/MyWebServer-master/bin/se/lnu/http/resources/inner/”.
4. In web broser enter “localhost:9000”.

**Expected Results:**

Web browser:



Console:



```
HTTPServerConsole x
"C:\Computer Science\Programming\Java\JDK\7\jdk1.7.0_80\bin\java.exe" ...
HTTP Server object constructed
HTTP Server started
Accept
|

ClientThread 2 served file : works2.png
ClientThread stopped nr: 1
ClientThread stopped nr: 2
Accept
ClientThread started nr: 3
ClientThread 3 served file : index.html
ClientThread 3 served file : works.png
ClientThread 3 served file : works2.png
ClientThread stopped nr: 3
```

**Actual Results:** **Pass**





## 3.2 Test Case 2: Start Server with Taken Socket Port Number

**Use Case:** UC 1.2. Start Server

**Requirement:** Req3

**Scenario:** System asks for socket port number and shared resource container.

The web server could not be started due to the socket was taken.

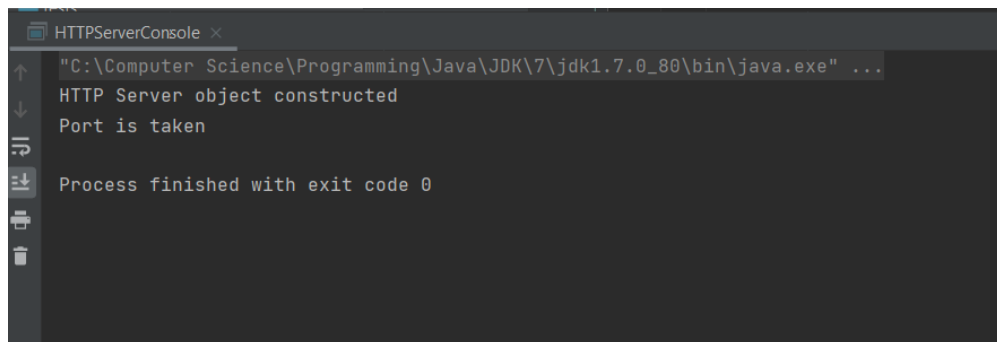
**Precondition:** The server is not deployed yet

**Test Steps:**

1. Set Project JDK to java “version 1.7” and Project language level to “7 – Diamonds, ARM, multi-catch etc”.
2. Edit Configuration for the class “HTTPServerConsole”, and set Program Arguments as “80  
C:/Github/Repository/2DV610/MyWebServer-master/bin/se/lnu/http/resources/inner”.
3. In terminal enter “ java se.lnu.http.HTTPServerConsole 80  
~/MyWebServer-master/bin/se/lnu/http/resources/inner”.
4. In web broser enter “localhost:80”.

**Expected Results:**

Console:



```
HTTPServerConsole x
"C:\Computer Science\Programming\Java\JDK\7\jdk1.7.0_80\bin\java.exe" ...
HTTP Server object constructed
Port is taken
Process finished with exit code 0
```

**Actual Results:** Pass



### 3.3 Test Case 3: Stop Server Successfully

**Use Case:** UC 2. Stop Server

**Requirement:** Req3

**Scenario:**

1. Starts when a user wants to stop the server.
2. The system stops the web server and presents that the webserver has been stopped.

**Precondition:** A web server has been started.

**Test Steps:**

1. Type “stop” in the console.
2. Navigate to web server.

**Expected Results:**

Web browser:

localhost:9000



## This site can't be reached

**localhost** refused to connect.

Try:

- Checking the connection
- [Checking the proxy and the firewall](#)

ERR\_CONNECTION\_REFUSED

Reload

Console:



```
HTTPServerConsole x
ClientThread 1 served file : works.png
ClientThread 2 served file : works2.png
ClientThread stopped nr: 1
ClientThread stopped nr: 2
stop
HTTP Server Accept thread stopped
HTTP Server stopped

Process finished with exit code 0
```

**Actual Results:** **Pass**



### 3.4 Test Case 4: Request Shared Resource Not Found

**Use Case:** UC 3. Request Shared Resource

**Requirement:** Req2, Req3

**Scenario:** The system delivers the shared resource to the browser and a success message is written to the access log. The shared resource cannot be found.


**Precondition:** A web server has been started.

**Test Steps:**

1. Remove the secret.html file.
2. Navigate to web server.

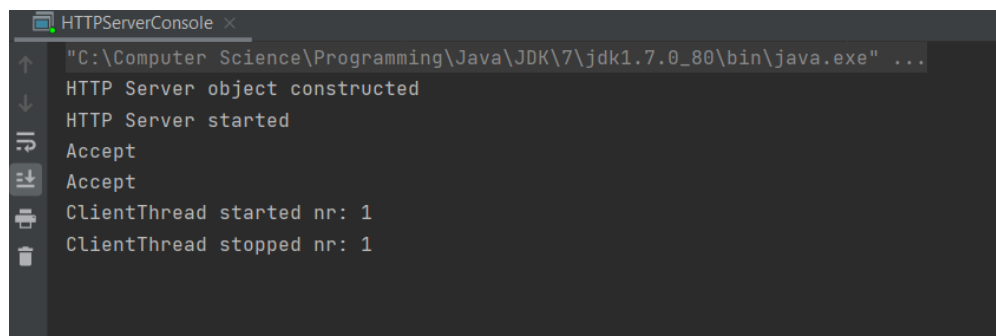
**Expected Results:**

Web browser:

 localhost:9000

# 404 Not found

Console:



```
HTTPServerConsole x
"C:\Computer Science\Programming\Java\JDK\7\jdk1.7.0_80\bin\java.exe" ...
HTTP Server object constructed
HTTP Server started
Accept
Accept
ClientThread started nr: 1
ClientThread stopped nr: 1
```

**Actual Results:** **Pass**



### 3.5 Test Case 5: GPL-2.0

**Requirement:** Req4

**Scenario:** The source code should be released under GPL-2.0

**Precondition:** A web server has been started.

**Test Steps:**

1. Check LICENSE file.

**Expected Results:**

GPL-2.0 (General Public License, version 2)

**Actual Results:** **Fail**

MIT (Massachusetts Institute of Technology License)

```
The MIT License (MIT)

Copyright (c) 2014 Daniel Toll

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
SOFTWARE.
```