Customer Requirements Specification

(Lastenheft)

Author: Lucas Kaczynski

(TINF20C, SWE I Praxisprojekt 2021/2022)

Project: Websockets in a LwIP HTTP Server

Customer: Rentschler & Holder
Rotebühlplatz 41
70178 Stuttgart

Supplier: Team 4: Laura Reeken, inf20051@lehre.dhbw-stuttgart.de
Benjamin Esenwein, inf20074@lehre.dhbw-stuttgart.de
Yannis Plaschko, inf20093@lehre.dhbw-stuttgart.de
Maximilian Meier, inf20084@lehre.dhbw-stuttgart.de
Lucas Kaczynski, inf20147@lehre.dhbw-stuttgart.de
Isabel Schwalm, inf20085@lehre.dhbw-stuttgart.de
Rotebühlplatz 41
70178 Stuttgart

Version	Date	Author	Comment
0.1	20.09.2021	Lucas Kaczynski	Document created. Formulated the general components of the project using the CRS template. Cleaned up the goal description. Added a Product Environment description. Improved spelling and grammar. Added English translation.
1.0	05.11.2021	Lucas Kaczynski	Content review. Document clean-up

CONTENTS

1. Goal

lwIP is a fully functional TCP/IP stack focusing on consuming as few resources as possible. It is written In the C programming language and usable for free.
Because of its high efficiency, lwIP is used for example by Intel, Xilinx and Analog Devices in embedded systems [1]. For example, lwIP can be used on an ESP8266 [2].
The goal of this project is to address the architectural deficiencies of patch "#9525 (httpd: add websocket support)" [3] in coordination with the project community. This experimental base is to be improved and brought through the approval process in the open-source project.
Furthermore, a demo server is to be designed and implemented in a virtual environment under Windows.
For demonstration and testing purposes of the features, a GUI-based performance test client shall be designed and implemented.

2. Product Environment

lwIP is used in embedded systems that act as web servers. The client makes a communication request to the web server and lwIP establishes a connection between the two communication endpoints. These communication endpoints are also called sockets. Data can be sent and received by both parties. However, this procedure causes delays because the client must first send a request to the server before the web page can be sent to the client [4]. The basic functionality can be seen in figure 1.

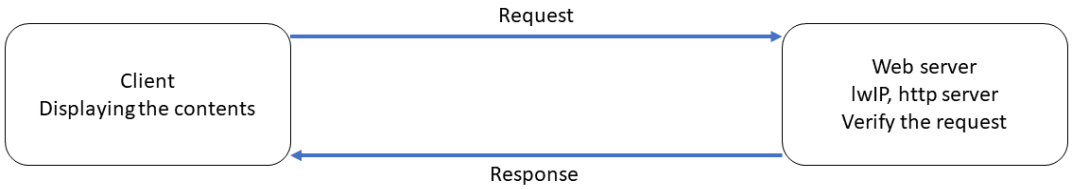

Figure 1. lwIP basic functionality.

Building on this, websockets extend this connection and make it bidirectional. Data can be sent and received between the sockets simultaneously. A client gets the web page immediately when it opens a connection to the server.
The connection between the server and the client remains open. Thus, the server can send information to the client without the client needing to request it [4]. This process can be seen in figure 2.

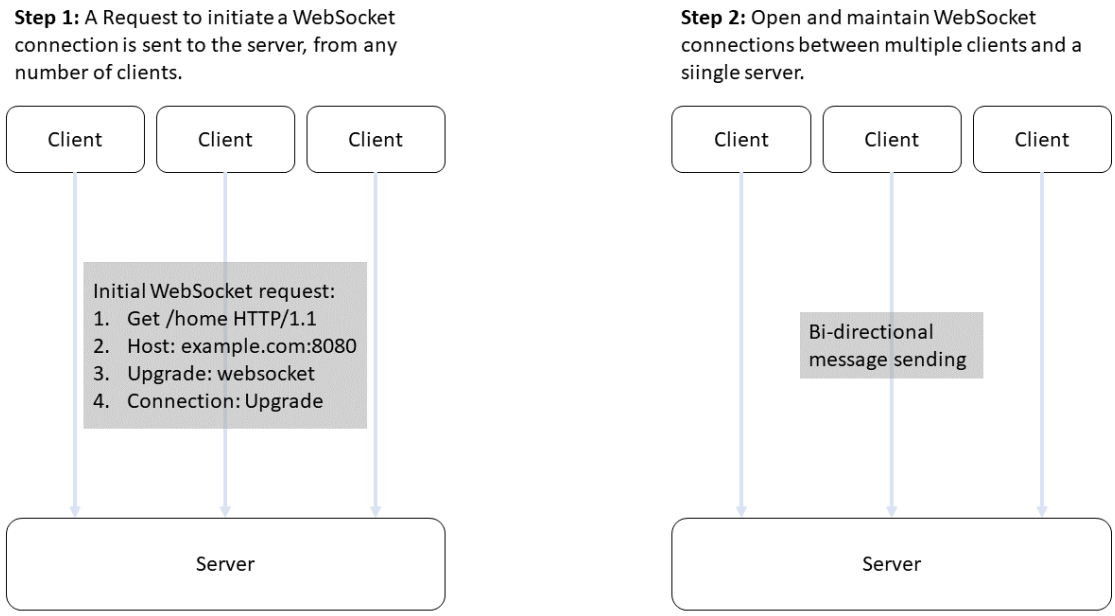

Figure 2. Websocket Protocol.

3. Product Usage

The users are looking for a stable, established TCP/IP stack, which is continuously developed further. lwIP fulfills these conditions. So far, however, lwIP does not support websockets, which is why websocket support is to be added in this project [3]. For this purpose, the documentation of lwIP is supplemented with instructions on the use of websockets and code examples. The user can set up a demo server running a rudimentary HTTP server whose communication is realized using a websocket. The HTTP server can be accessed through a simple GUI and allows the user to check the features of the HTTP server.

3.1. Business Processes

This is an open-source project. Accordingly, profit is not the primary interest of this project. The project is developed with the aim to represent a time and cost saving for companies and developers.

3.2. Use Cases

Related Business Process:	Prozess-ID: UC01
Use Cases Objective:	Customer is developing a CPU and wants to ensure support for TCP/IP and Websockets.
System Boundary:	lwIP allows the hardware to use TCP/IP.
Precondition:	The user can find and download lwIP via a conventional Internet browser. Setting up the software is adequately documented, and users can do it themselves.
Postcondition on success:	lwIP is executable on the CPU of the user.
Beteiligte Nutzer:	Role name: CPU developer
Triggering Event:	The developer gets the order to extend the functionality of an embedded system with TCP/IP and websocket support.

Related Business Process:	Prozess-ID: UC02
Use Cases Objective:	Customer works with microcontrollers (e.g. ESP8266) as a hobby and wants to build a TCP/IP stack on top of it.
System Boundary:	lwIP allows the hardware to use TCP/IP.
Precondition:	The user can find and download lwIP via a conventional Internet browser. Setting up the software is adequately documented, and users can do it themselves.
Postcondition on success:	lwIP is executable on the user's microcontroller.
Beteiligte Nutzer:	Role name: Hobby developer
Triggering Event:	The developer gets the order to extend the functionality of an embedded system with TCP/IP and websocket support.

4. Other Product Characteristics

lwIP was developed for use in embedded systems. Accordingly, lwIP requires minimal RAM and ROM. The attractiveness of this product clearly lies in its open-source availability. This promotes the dissemination and further development of the product. However, the further development of lwIP will not take place on an embedded system, but on Windows. Accordingly, a suitable development environment must be set up.

5. References

[1] A. Sarangi, S. MacMahon, and U. Cherukupaly, 'LightWeight IP Application Examples', 2014, Accessed: Nov. 05, 2021. [Online]. Available: https://www.semanticscholar.org/paper/LightWeight-IP-Application-Examples-Sarangi-MacMahon/46a49c7c0430240263f0496a7216fe7f6bf8f60b

[2] lujji, 'HTTP server with WebSockets on ESP8266', *lujji*. http://lujji.github.io/blog/esp-httpd/index.html (accessed Oct. 15, 2021).

[3] S. Kapanen, 'lwIP - A Lightweight TCP/IP stack - Patches: patch #9525, httpd: add Websocket support [Savannah]', Dec. 21, 2017. https://savannah.nongnu.org/patch/?9525 (accessed Nov. 05, 2021).

[4] 'Was ist WebSocket?', *IONOS Digitalguide*, Mar. 16, 2020. https://www.ionos.de/digitalguide/websites/web-entwicklung/was-ist-websocket/ (accessed Oct. 15, 2021).