

# WebSocket's im LwIP HTTP Server

---

Team 4

13.05.2022

# TEAMVORSTELLUNG

Name	Matrikelnummer	Projektrolle	eMail-Adresse
Lucas Kaczynski	9411266	Techn. Redakteur	inf20147@lehre.dhbw-stuttgart.de
Benjamin Esenwein	5655208	Produktmanager	inf20074@lehre.dhbw-stuttgart.de
Maximilian Meier	6020212	Leitender Entwickler	inf20084@lehre.dhbw-stuttgart.de
Yannis Plaschko	6645234	Testmanager	inf20093@lehre.dhbw-stuttgart.de
Laura Reeken	3811697	Projektleiterin	inf20051@lehre.dhbw-stuttgart.de
Isabel Schwalm	6038831	Systemarchitektin	inf20085@lehre.dhbw-stuttgart.de

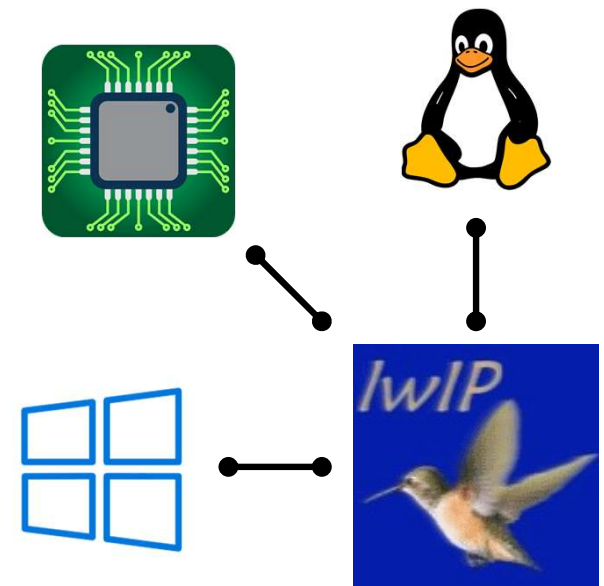


# INHALT

- Was ist IwIP?
- Master Use Case
- Systemarchitektur & Produktübersicht
- Funktionale und Nichtfunktionale Anforderungen
- Module
- Entwicklungsprozess
- Vorgehen beim Testen
- Testplan
- Rückblick Prototyp & Live Demo
- Fazit / Ausblick

# Was ist lwIP?

- kleine unabhängige Implementierung des TCP/IP-Protokolls
  - Zielsetzung
    - Reduzierung des Ressourcenverbrauchs
    - Gleichzeitige Bereitstellung eines vollwertigen TCP/IP-Protokolls
- Geeignet für Embedded Systems
- Lauffähig auf mehreren Betriebssystemen



# MASTER USECASE



Ziel: ● Websocket Support für lwIP implementieren  
● lwIP-Status Seite & REST-API implementieren



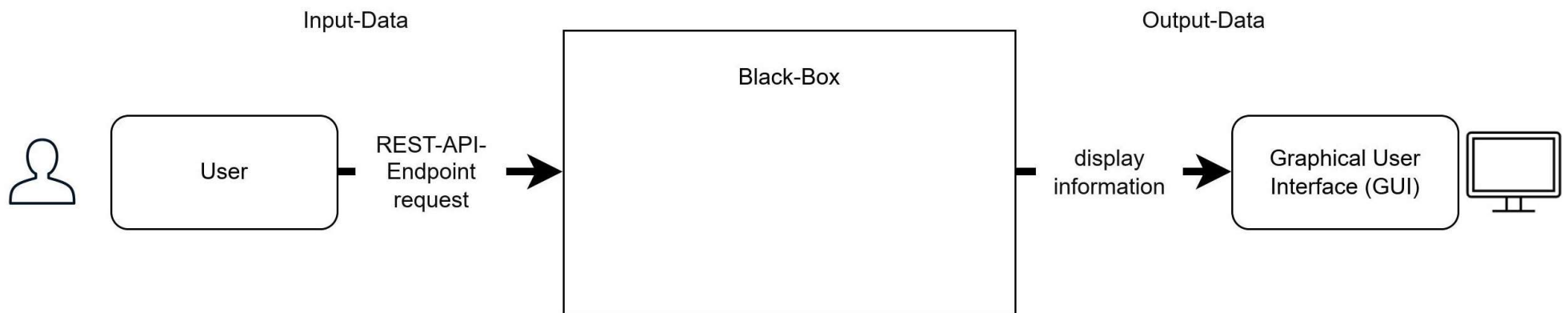
Aufgabe: Mängel des vorhandenen Patch #9525 beheben



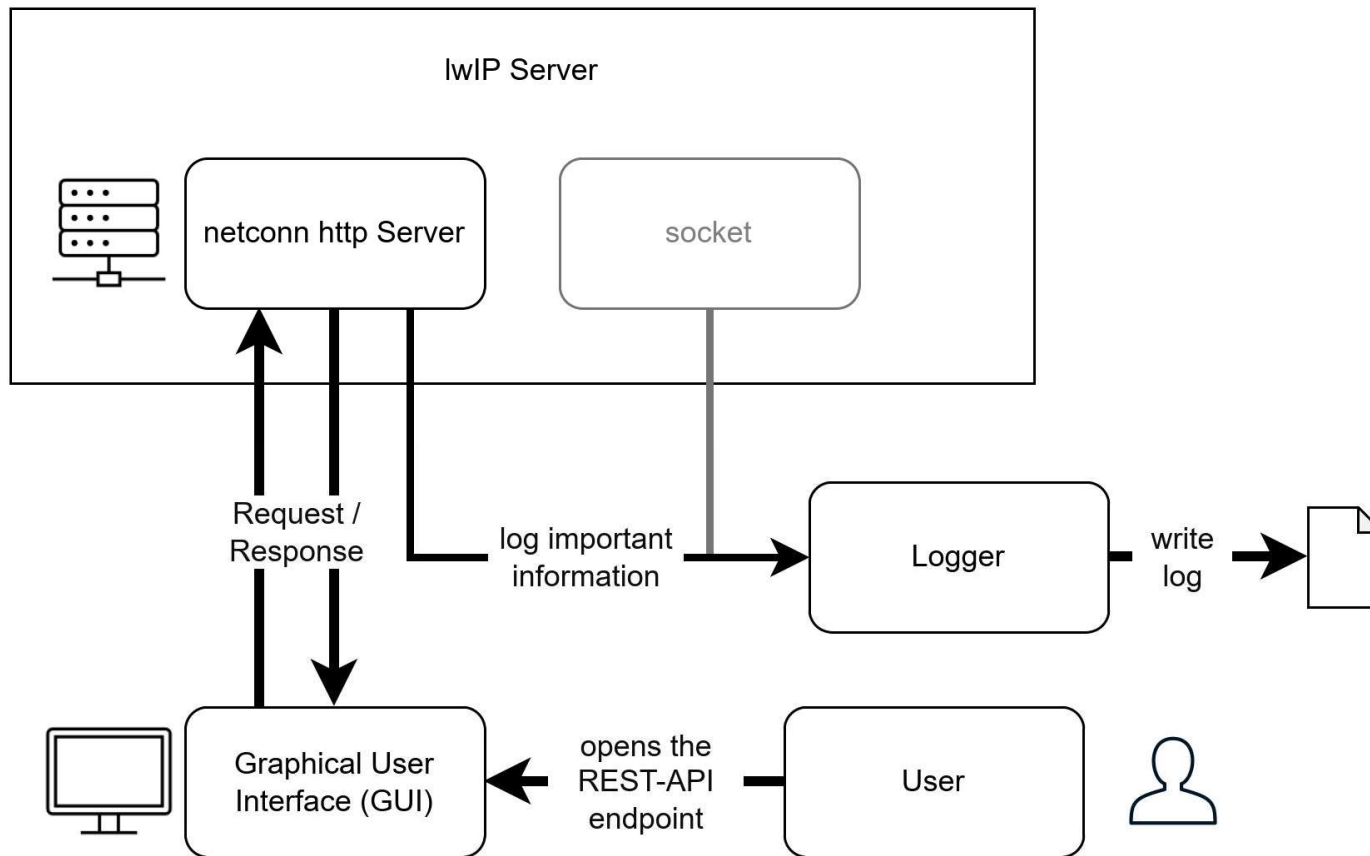
Use Case:

Beteiligte Nutzer: Entwickler von Embedded System  
Zielsetzung: ● Support für TCP/IP und Websockets  
● lwIP ist auf CPU lauffähig

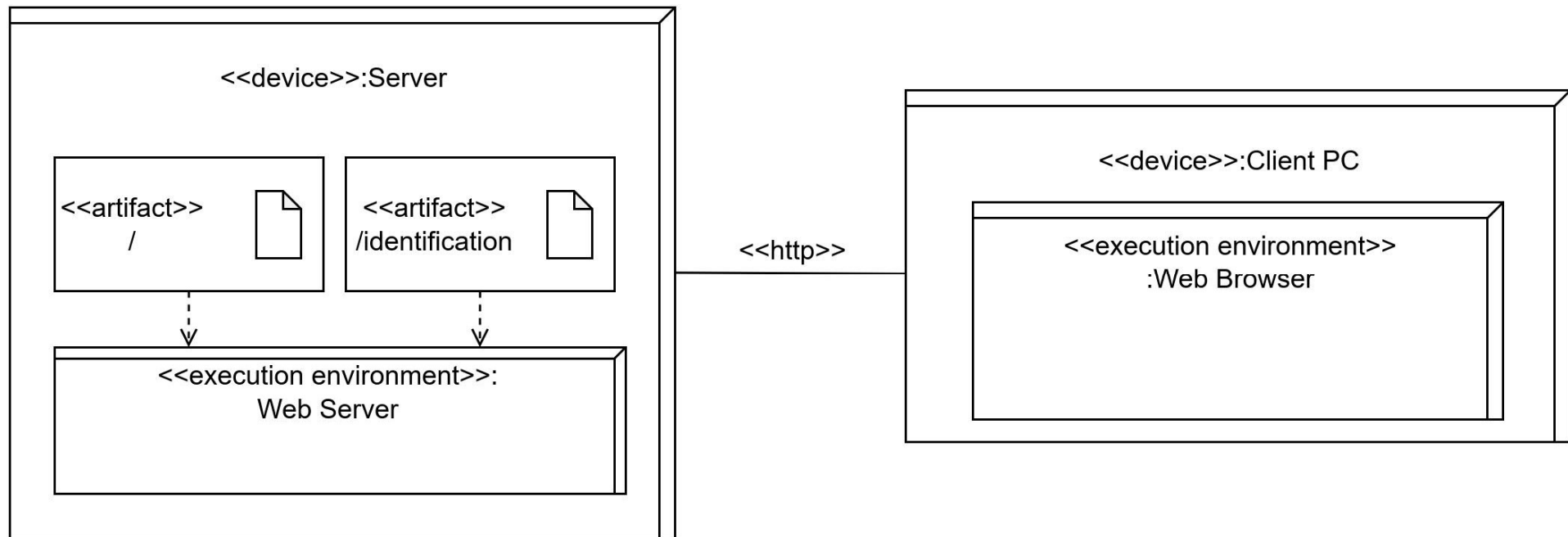
# SYSTEMARCHITEKTUR



# Produktübersicht



# Produktübersicht



Verteilungsdiagramm



# FUNKTIONALE ANFORDERUNGEN

/F10/ Extra Funktion für base64 encoding

/F20/ Implementation der WebSocket API

/F30/ Schreiben eines GUI-basierten Test-Client

/F40/ http Webserver für Testzwecke

# NICHTFUNKTIONALE ANFORDERUNGEN

/NF10/ Zusätzlich geschriebener Code sollte bei einem Minimum gehalten werden

/NF20/ Effizienter Code

/NF30/ Intuitive GUI für Test-Client

~~/NF40/ Bestehen des Approval Process~~ /NF41/ Informationsweitergabe

/NF50/ Executable Program

# Module

/MOD.001/      Graphical User Interface

/MOD.002/      HTTP-Server

/MOD.003/      WebSocket Server

# Entwicklungsprozess

- Visual Studio 2022
- WIN SDK 10.0
- Analyse & Einarbeitung Patch#9525
- Debugging
- Alternative WebSocket Server

libwebsockets.org



Bildquellen:

[https://logos.fandom.com/wiki/Microsoft\\_Visual\\_Studio#2021.E2.80.93present](https://logos.fandom.com/wiki/Microsoft_Visual_Studio#2021.E2.80.93present)

<https://cmake.org/>

[https://www.winpcap.org/docs/docs\\_412/html/](https://www.winpcap.org/docs/docs_412/html/)

[Npcap: Windows Packet Capture Library & Driver](#)

[libwebsockets.org lightweight and flexible C networking library](#)



# VORGEHEN BEIM TESTEN

- LwIP ist sehr umfangreich und komplex
  - Kein White-Box-Testing nicht möglich
- Black-Box-Testing
- GUI als POC daher kaum Testdaten
- POO aus Sicht des Users
- Testplanung
  - Verifikation der Lauffähigkeit
  - Erzeugen und Zerstören von Verbindungen

# Testplan

Test-ID	Req. - ID	Functionality
1	LF60: Call API root	This test verifies that calling the Root Endpoint returns a sample website
2	LF70: Call API /something	This test case verifies that calling an undefined Endpoint results in an error Message

## 11.2.1 <TC-002-001> (Call API Root)>

Testcase ID:	TC-002-001	
Testcase Name:	Call API Root	
Req.-ID:	LF60	
Description:	This test case verifies that calling the Root Endpoint returns a sample website.	
Test Steps:		
Step	Action	Expected result
1	Run the LwIP HttpServer	Application starts without problems.
2	Open the Browser and type the IP of the Server in the Search bar and hit enter	A Site should become visible with the Headline "LwIP Test Application", a Text block and a link to "/identification";
Testdata:	TD-002-001	
Dataset	Input File	Output File
1	X	X

# RÜCKBLICK AUF PROTOTYP

- Rückmeldung der lwIP-Community stand aus
- Code-Abhängigkeits-Fehler

Fehlerliste - Aktuelles Projekt (lwIP)

Aktuelles Projekt 64 Fehler 0 von 966 Warnungen 0 Mitteilungen Erstellen + IntelliSense

Code	Beschreibung	Projekt	Datei	Zeile
C2220	Warnung wird als Fehler interpretiert, es wurde keine object-Datei generiert.	lwIP	fsm.c	754
C2220	Warnung wird als Fehler interpretiert, es wurde keine object-Datei generiert.	lwIP	ipv6cp.c	1535
C2220	Warnung wird als Fehler interpretiert, es wurde keine object-Datei generiert.	lwIP	ppp.c	458
C2220	Warnung wird als Fehler interpretiert, es wurde keine object-Datei generiert.	lwIP	pppoe.c	776
C2220	Warnung wird als Fehler interpretiert, es wurde keine object-Datei generiert.	lwIP	utils.c	639
C2220	Warnung wird als Fehler interpretiert, es wurde keine object-Datei generiert.	lwIP	vj.c	412
C2220	Warnung wird als Fehler interpretiert, es wurde keine object-Datei generiert.	lwIP	httpd.c	537
C2220	Warnung wird als Fehler interpretiert, es wurde keine object-Datei generiert.	lwIP	snmp_core.c	239
C2220	Warnung wird als Fehler interpretiert, es wurde keine object-Datei generiert.	lwIP	snmp_msg.c	181
C2220	Warnung wird als Fehler interpretiert, es wurde keine object-Datei generiert.	lwIP	snmp_raw.c	92
C2220	Warnung wird als Fehler interpretiert, es wurde keine object-Datei generiert.	lwIP	snmp_threadsync.c	212
C2220	Warnung wird als Fehler interpretiert, es wurde keine object-Datei generiert.	lwIP	snmp_mib2_system.c	270
C2220	Warnung wird als Fehler interpretiert, es wurde keine object-Datei generiert.	lwIP	snmp_mib2_tcp.c	195

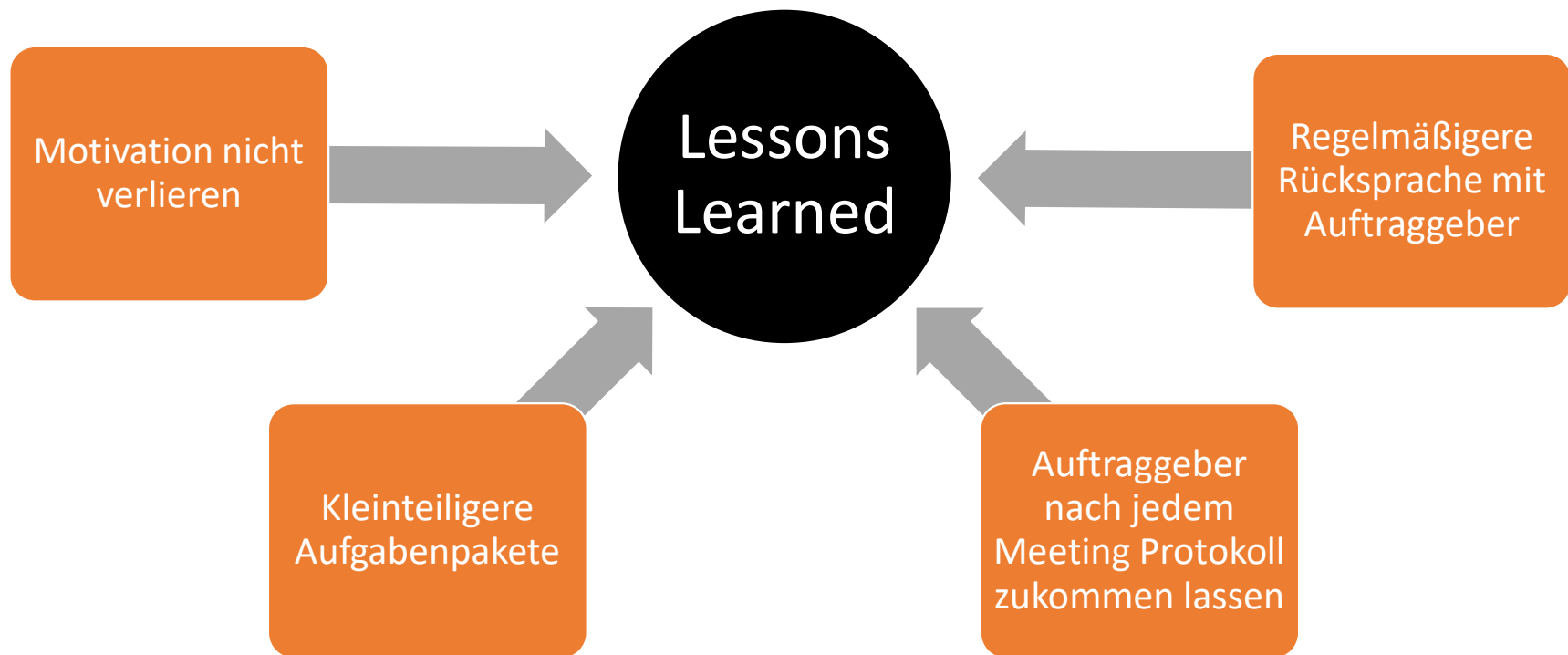


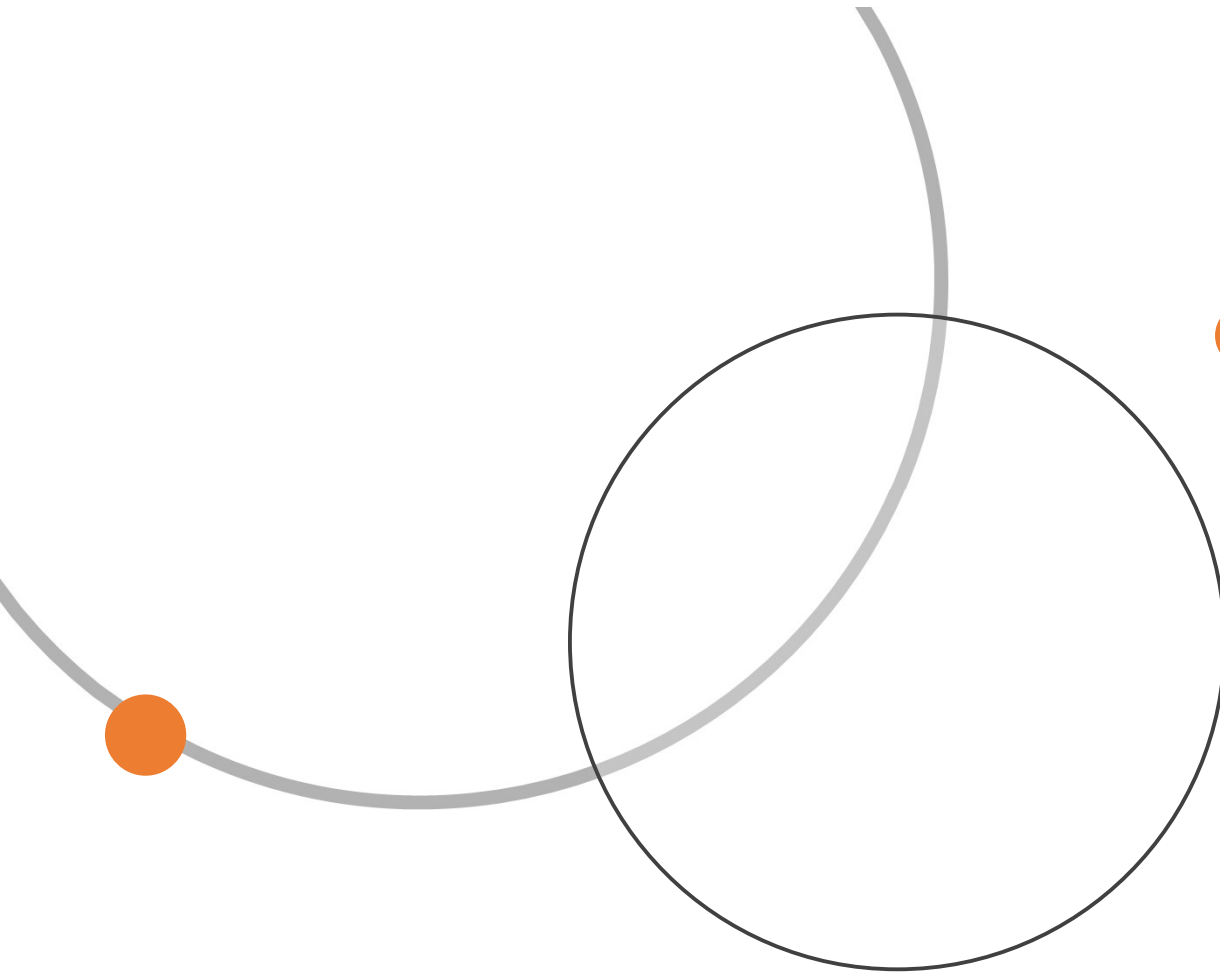
[Anwendung öffnen](#)

[JSON Validator](#)



# FAZIT





- Websockets  
im LwIP  
HTTP  
Server

---

Team 4

13.05.2022