Programmer friendly communication framework

1st Lilo Zobl
Botball Team FrenchBakery
HTL Anichstraße
Innsbruck, Austria
lzobl@tsn.at

2nd Matteo Reiter
Botball Team FrenchBakery
HTL Anichstraße
Innsbruck, Austria
mareiter@tsn.at

***Abstract*—This document will focus on an important topic in software development: Network communication. It will provide an insight into why what network communication is needed for and how it is typically realized. Additionally it will cover problems and annoyances with currently available communication frameworks for certain applications as well as introduce concepts and a possible solution for the beforementioned.**

I. Introduction

When it comes to computer ... Why do we need communication? What for?

IP Networks erwähnen

TCP/UDP Erwähnen und unterschied (Stream vs. Datagram).

wir behandeln hier nur direkte punkt-zu-punkt verbindungen

A. Communication frameworks

... Why do we need communication frameworks?

using TCP/UDP directly is tedious for many modern application, ...

Job of cf/library?

- abstrahieren
- benutzerfreundlichkeit bzw programmiererfreundlichkeit
- added features for specialized use cases

II. State of the Art

- TCP & UDP (mostly TCP) for low-level, very performant systems
- WebSockets: Implements an additional protocol on top of TCP re-introducing the concept of messages. For Web applications already using HTTP(s)
- SocketIO: Builds on top of WebSocket, allows sending javascript events over the network, adding event listeners/emitters and allow sending arbitrary data.

III. Problems with current offer

This section builds upon the last mentioned protocol SocketIO.

No reassurance of data validity. before using incoming data, one ideally has to check that the structure and type of the received data matches what is expected and handle the error if that isn't the case. This is tedious to do manually for every single event. Additionally since SocketIO allows sending arbitrary data, it is easy to accidentally send data in an incorrect structure, such as accidentally making a typo in an attribute name.

IV. Solution and Design Overview

V. Implementation

VI. Results and Applications

VII. Limitations

VIII. Conclusion

List of Figures

References

[1] Proof for NordPass study, URL:https://tech.co/password-managers/how-many-passwords-average-person