Documentatie

Pop Vlad Daniel

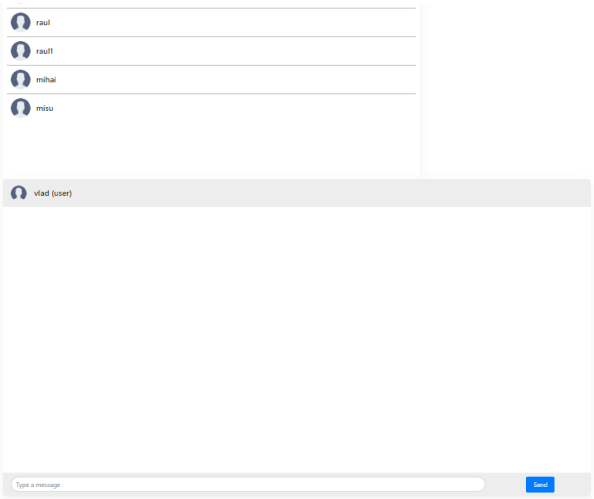
November 2022

1 Content

2 Introduction

An online chat connects users with administrator. The user is able to talk to the administrator, for problems about his device measurements. After the admin is online, he can chat with multiple users and help them solve their problems

3 How the app looks like



This is the chat page, from the administrator perspective where he can chat with multiple users at once. The user chat can only access the administrator for troubleshooting.

4 Implementation

Node.js lets developers use JavaScript to write command line tools and for server-side scripting. The functionality of running scripts server-side produces dynamic web page content before the page is sent to the user's web browser. Consequently, Node.js represents a "JavaScript everywhere" paradigm, unifying web-application development around a single programming language, rather than different languages for server-side and client-side scripts.

The project is build using node.js for the backend part, also using typescript because of the fact that solves a lot of

errors by using types.
Using mongoDB and more important, using mongo atlas, our database is already located in the cloud, so we do not have to deploy it.
For frontend was used angular, being one of the best frameworks for this kind of job.

Socket.IO is an event-driven library for real-time web applications. It enables real-time, bi-directional communication between web clients and servers. It consists of two parts: a client-side library that runs in the browser, and a server-side library for Node.js. Both components have a nearly identical API.

Socket.IO primarily uses the WebSocket protocol with polling as a fallback option,[3] while providing the same interface. Although it can be used simply as a wrapper for WebSockets, it provides many more features, including broadcasting to multiple sockets, storing data associated with each client, and asynchronous I/O.

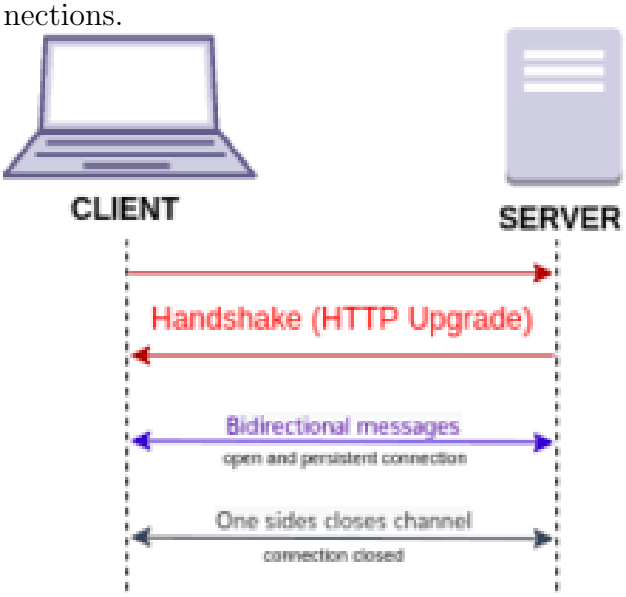
5 Websockets

WebSockets are a technology that enables real-time, bidirectional communication between a client and a server over a single, long-lived connection. They were first introduced in 2009 as part of the HTML5 standard, and have since become a popular choice for building real-time web applications such as chat apps, online gaming, and collaborative tools.

One of the main advantages of WebSockets is the ability to establish a full-duplex communication channel between the client and server. This means that the client and server can send messages to each other simultaneously, without having to wait for a response. This enables low-latency

communication, and allows for real-time data exchange between the client and server.

WebSockets also improve performance by eliminating the need for multiple connections between the client and server. With traditional web communication methods such as HTTP, the client would have to open a new connection for each request and response. With WebSockets, the connection is established once and remains open throughout the duration of the session, reducing the overhead and latency associated with creating and closing connections.



6 Deploy

The deploy was made with docker
Docker can package an application and its dependencies in a virtual container that can run on any Linux, Windows, or macOS computer. This enables the application to run in a variety of locations, such as on-premises, in public (see decentralized computing, distributed computing, and cloud computing) or private cloud. When running on Linux, Docker uses the resource isolation features of the Linux kernel (such as cgroups and kernel namespaces) and a union-capable file system (such as OverlayFS) to allow containers to run within a single Linux instance, avoiding the overhead of starting and maintaining virtual machines. Docker on macOS uses a Linux virtual machine to

run the containers.

7 Conclusion

In conclusion, a chat application that utilizes WebSockets allows for real-time, bidirectional communication between clients and servers. This technology enables faster and more efficient messaging compared to traditional HTTP requests, making it an ideal choice for applications such as online chats, multiplayer games, and other forms of real-time communication. Overall, the use of WebSockets in a chat application can greatly enhance the user experience by providing near-instantaneous messaging.