Secure & Scalable WebRTC-Based Communication Platform

School of Computer Application

BCA Hons.

Agenda

Topics Covered

1. Team & WrokFlow
2. Intro
3. Features
4. Project Discussion
5. Future Scope
6. Conclusion

Team Members

Name: Mohit Lamba
Reg: 12224201

Name: Chandra Bhanu Satapathy
Reg: 12223157

Name: Vikramjit Singh Gill
Reg: 12214506

Reg: Aman Kumar
Reg: 12210428

Name: Sahil Kumar Sharma
Reg: 12204019

Core Team Responsibility

Dev folks:

Vikramjit Singh Gill - (Frontend Design, Handlers, rooms, DB, API intrg.)

Sahil Kumar Sharma - (Design Pattern, Handlers, Stream, Chat, API intrg.)

Aman Kumar - (Frontend, Handlers, Authentication, API intrg.)

Chandra Bhanu Satapathy-(Design Pattern, Networking, EC2, research)

Mohit Lamba- (UI/UX, Design Pattern, Networking, VPC)

What is WebRTC ?

WebRTC (Web Real-Time Communication) is a technology that allows audio, video, and data sharing directly between web browsers without needing extra software or plugins. It enables real-time communication for video calls, chats, and file sharing over the internet.

Types of WebRTC

- Audio Communication – Real-time voice calls between users.
- Video Communication – Real-time video calls or conferencing.
- Data Communication – Peer-to-peer data sharing like chat messages or file transfers.

Applications in WebRTC

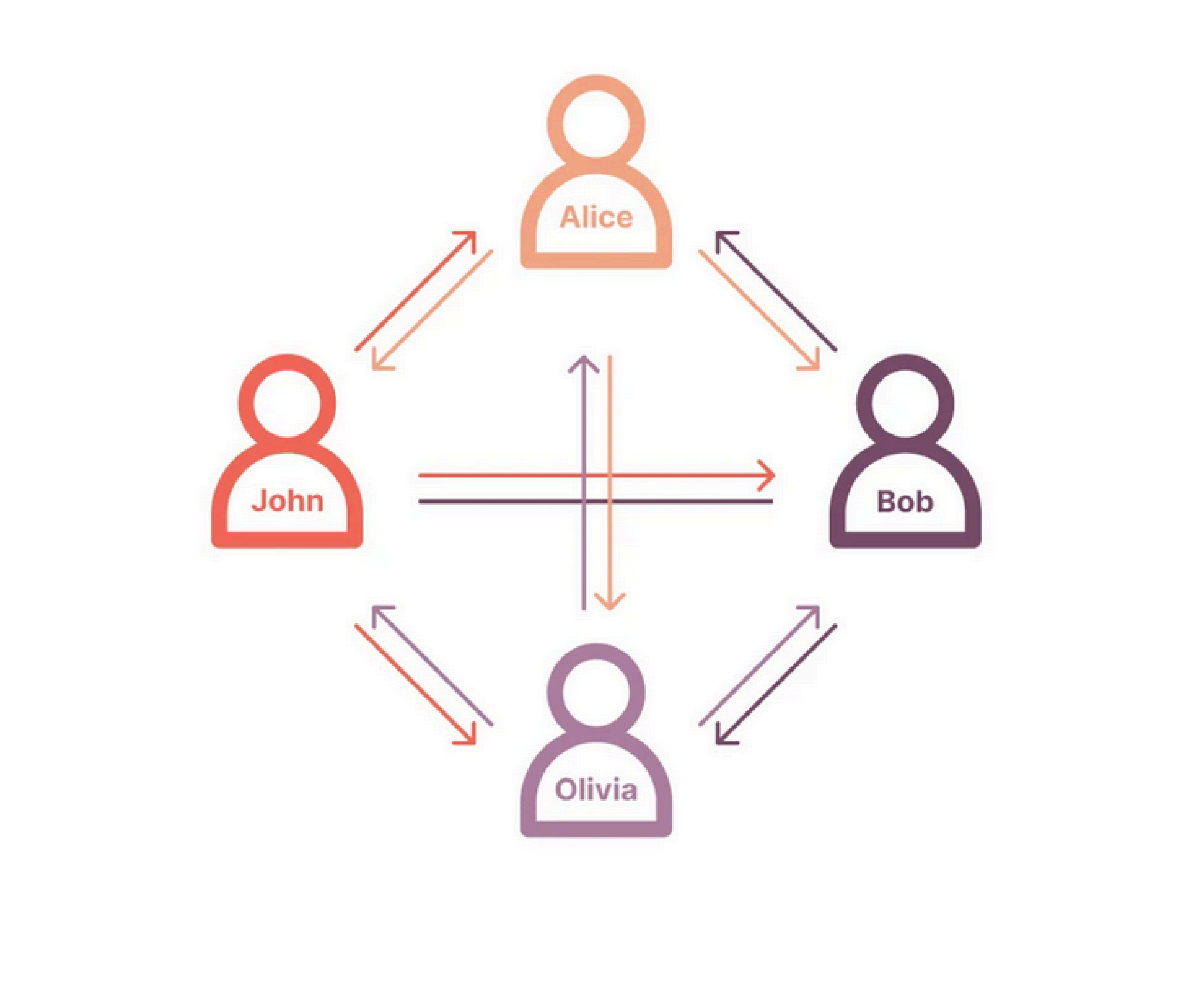
- Google Meet – For real-time video and audio conferencing.
- Discord – For voice, video, and screen sharing in gaming and communities.
- Facebook Messenger – For video calls directly through the browser or app.

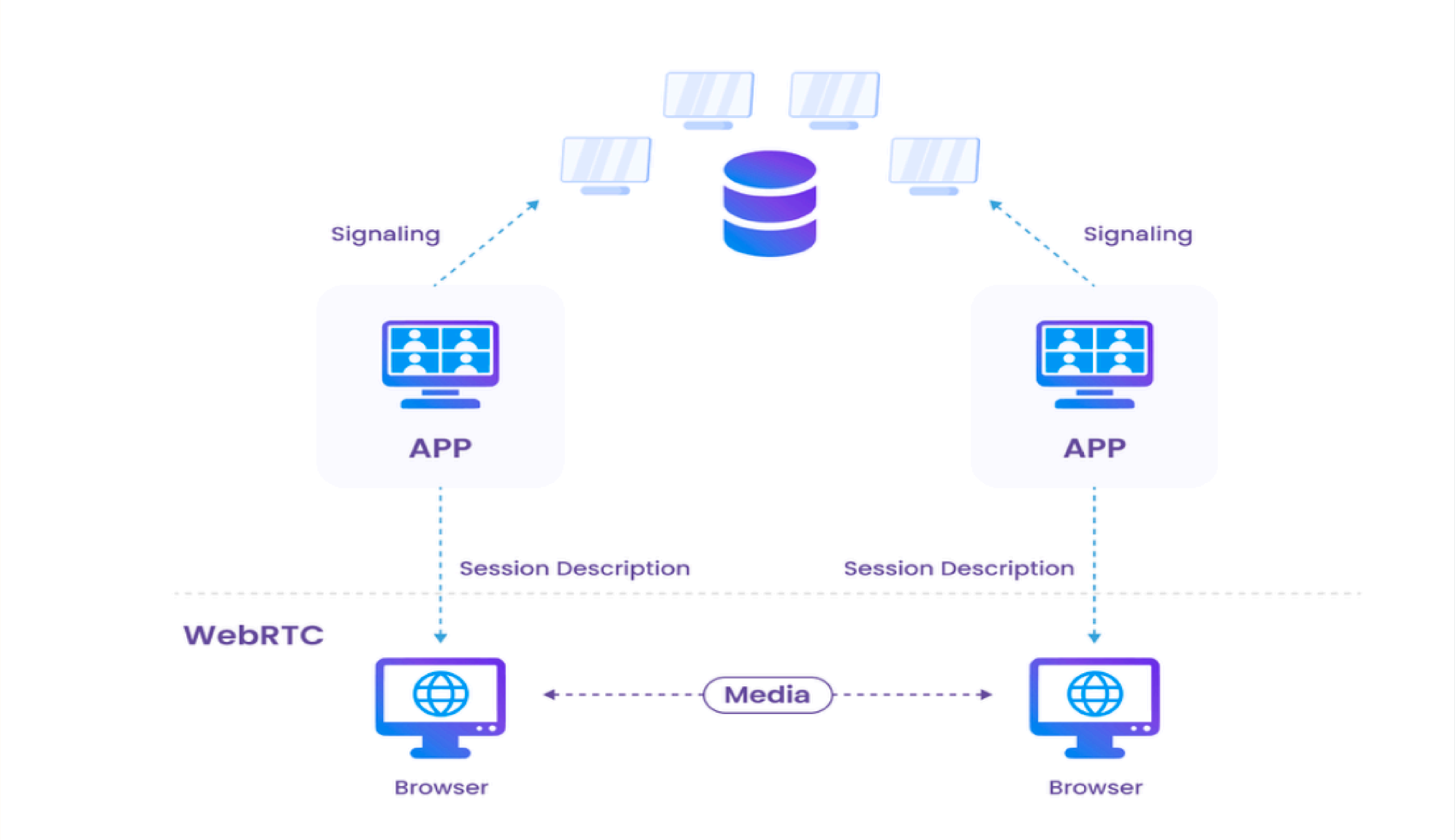
Our Project

- TURN – Ensures reliable media relay behind strict NAT/firewalls.
- GoLang – Lightweight backend for efficient signaling and performance.
- Scalable – Designed to handle multiple concurrent connections smoothly.
- Authentication – Secures user access and session integrity.
- Testing – Thorough validation for reliability and bug-free communication.

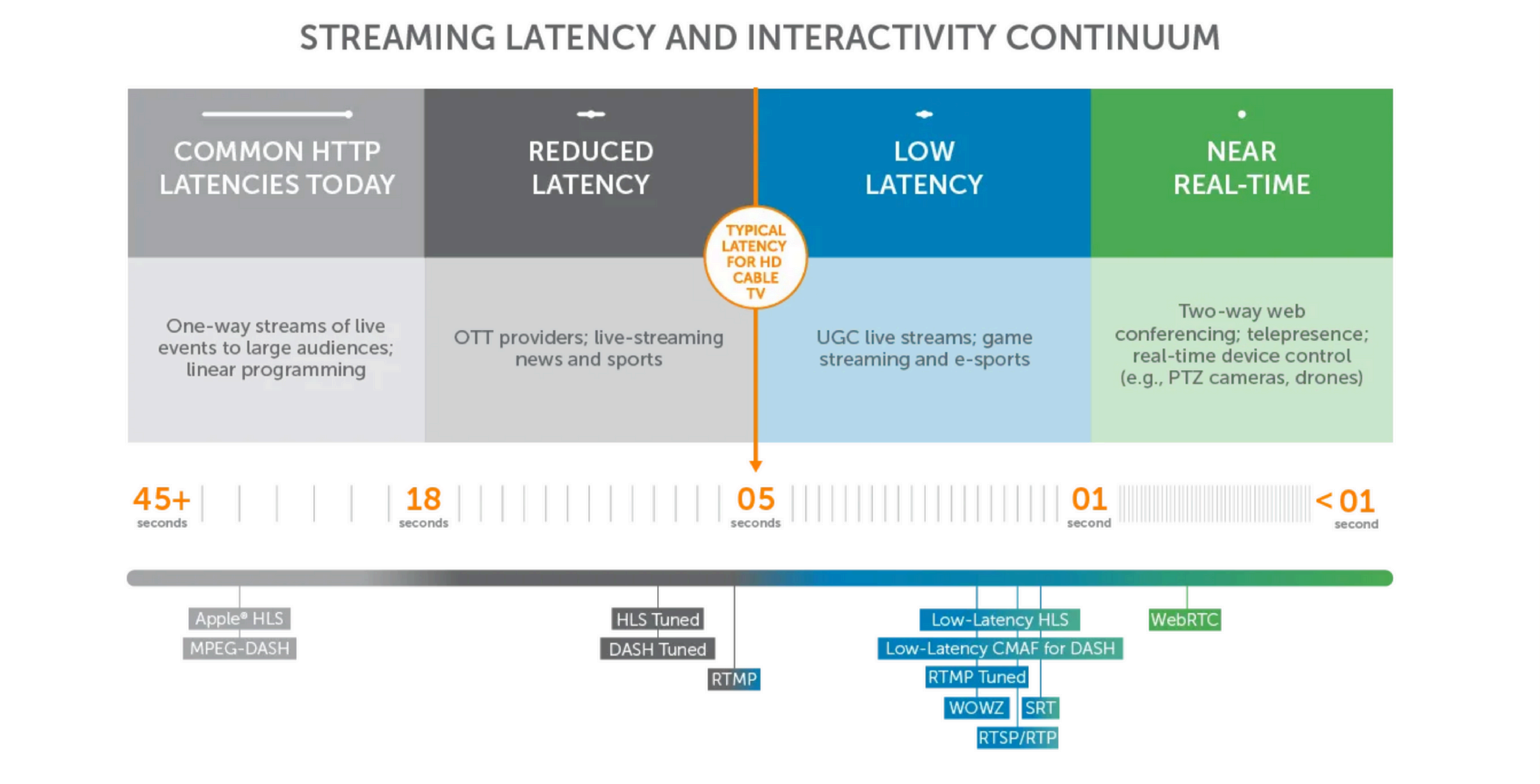
Real Time Communication Over Browser

webRTC

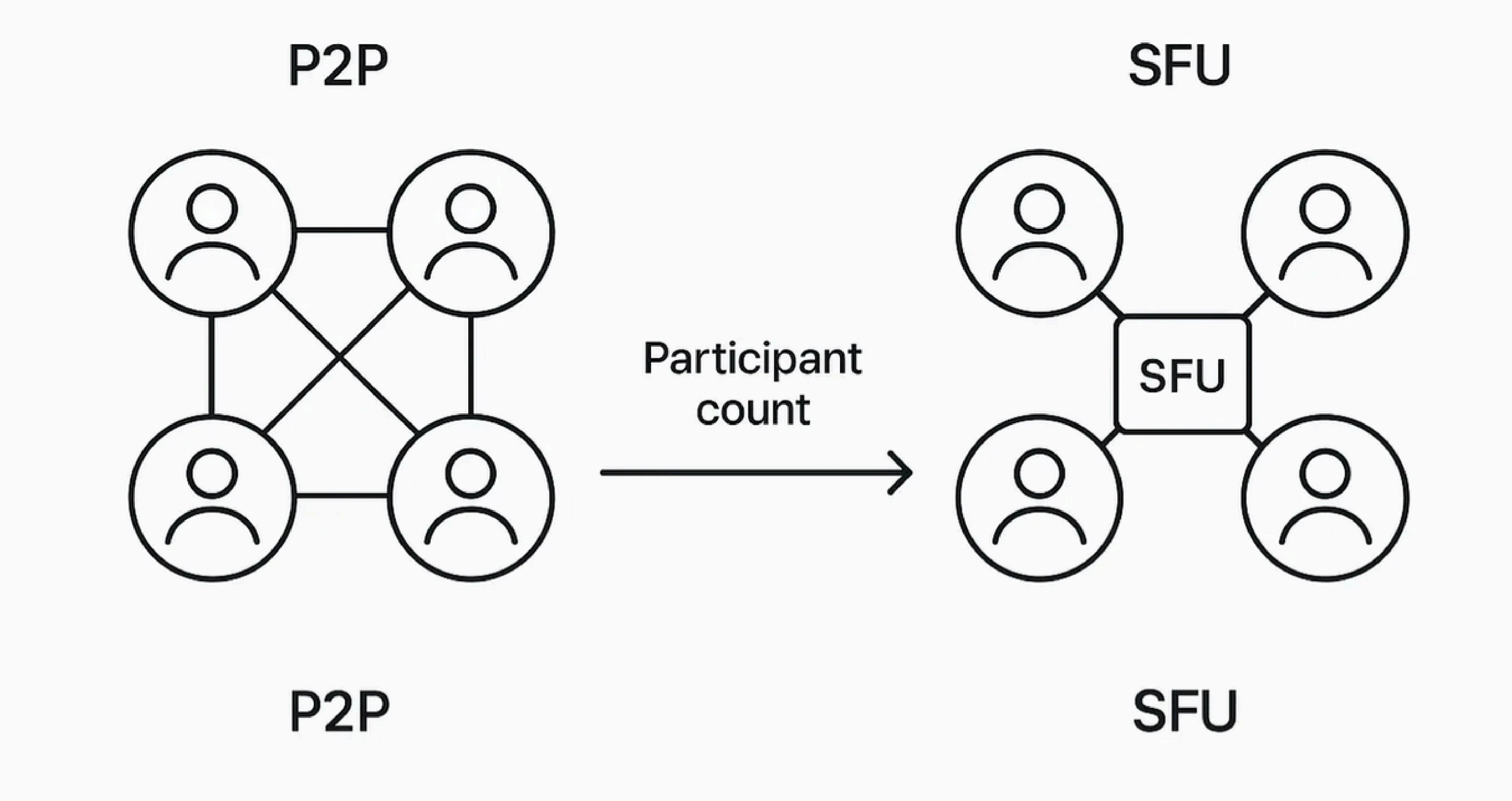




Technology Comparison



Architecture Design: Hybrid Approach Enable Scalability

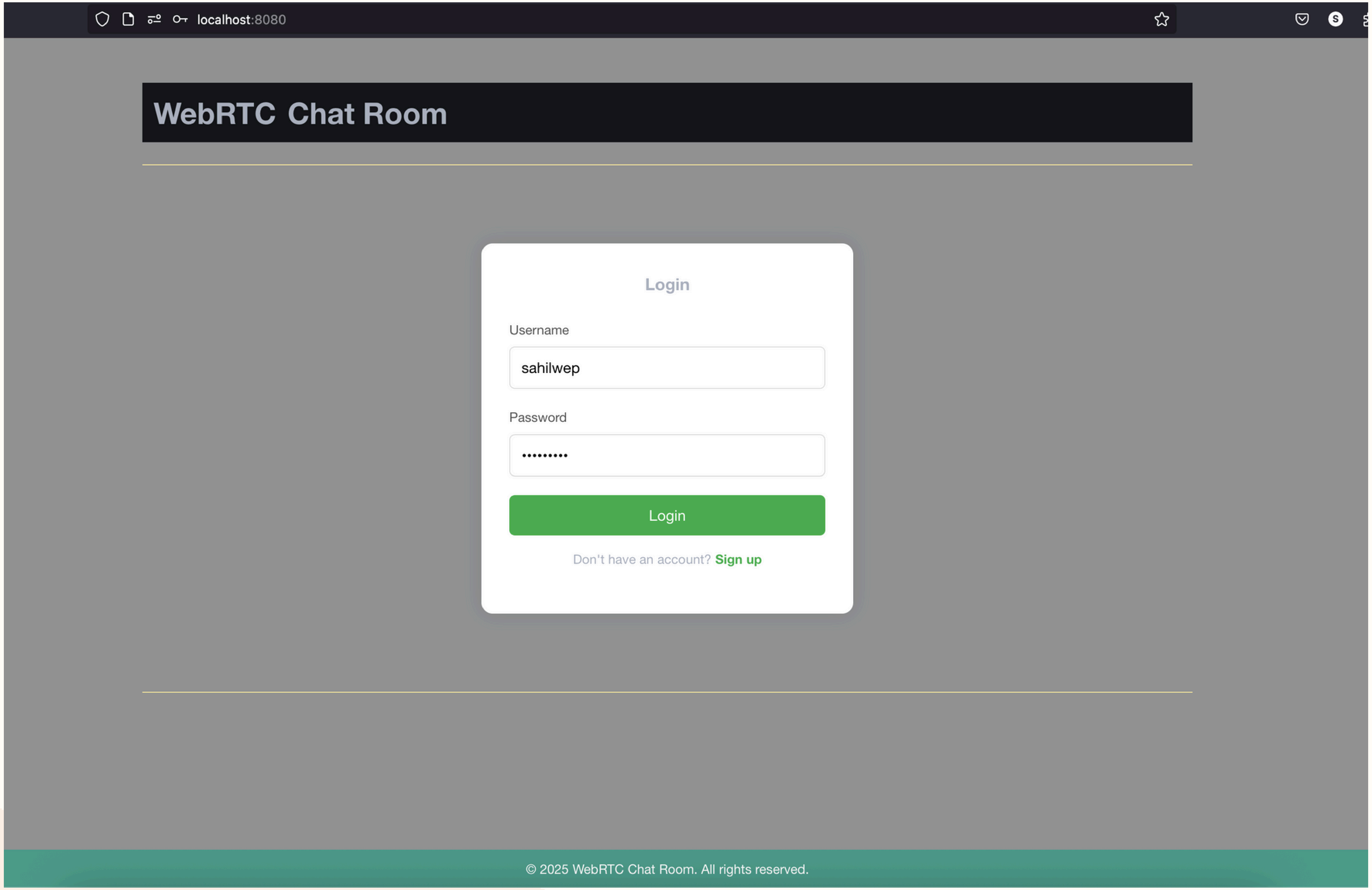


In the hybrid approach, different architectures are used depending on the number of participants in the call. Initially, **P2P** (Peer-to-Peer) is used for 1-1 calls, and as more participants join the call, the architecture switches to **SFU** (Selective Forwarding Unit) to accommodate the growing number of participants.

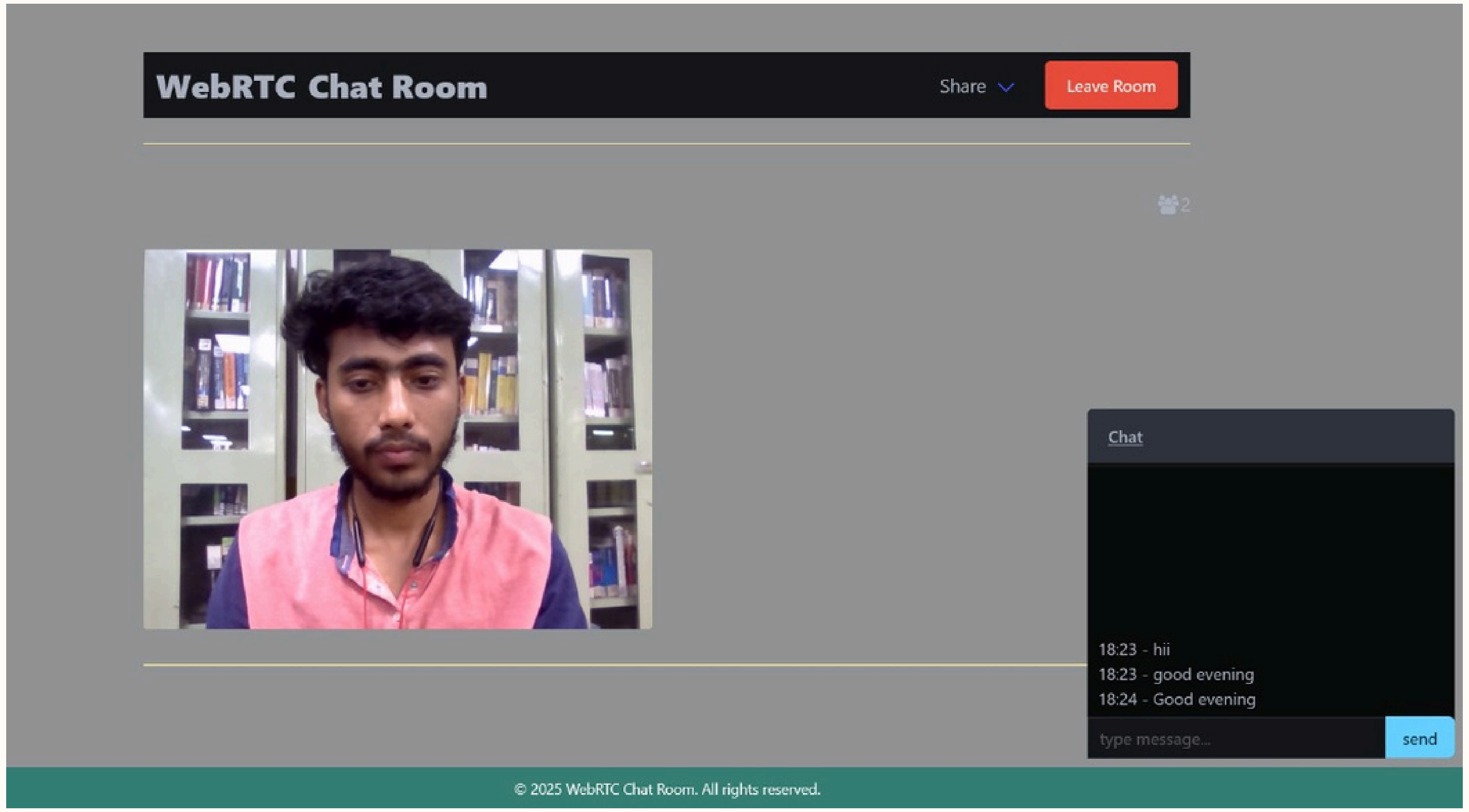
Fire up Project

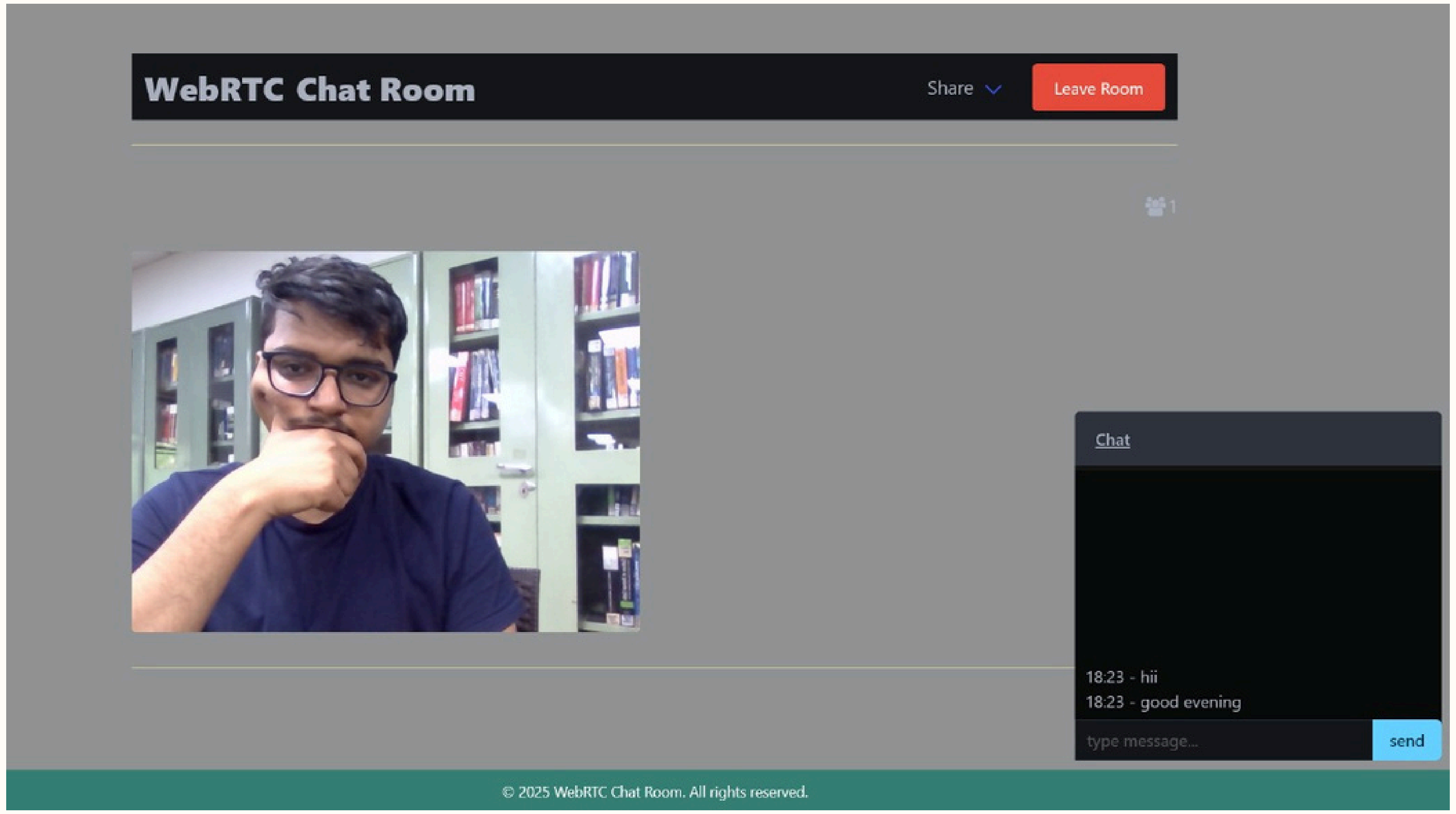
```
main                                                    ⌥⌘1

sahilwep~$ go run cmd/main.go

 ┌───────────────────────────────────────────────────┐
 │                 Fiber v2.52.6                     │
 │               http://127.0.0.1:8080               │
 │       (bound on host 0.0.0.0 and port 8080)       │
 │                                                   │
 │ Handlers ............ 29  Processes ........... 1 │
 │ Prefork ....... Disabled  PID ............. 39173 │
 └───────────────────────────────────────────────────┘

22:34:49 | 200 |     912.125µs | 127.0.0.1 | GET | / | -
22:34:49 | 200 |     6.68725ms | 127.0.0.1 | GET | /sw.js | -
22:34:49 | 404 |     111.583µs | 127.0.0.1 | GET | /favicon.ico | Canno
t GET /favicon.ico
22:34:54 | 302 |      14.708µs | 127.0.0.1 | POST | /login | -
22:34:54 | 200 |     210.625µs | 127.0.0.1 | GET | /welcome | -
22:34:54 | 200 |        10.5µs | 127.0.0.1 | GET | /sw.js | -
22:34:54 | 404 |     121.417µs | 127.0.0.1 | GET | /favicon.ico | Canno
t GET /favicon.ico
22:34:56 | 302 |      80.417µs | 127.0.0.1 | GET | /room/create | -
```

Login Page

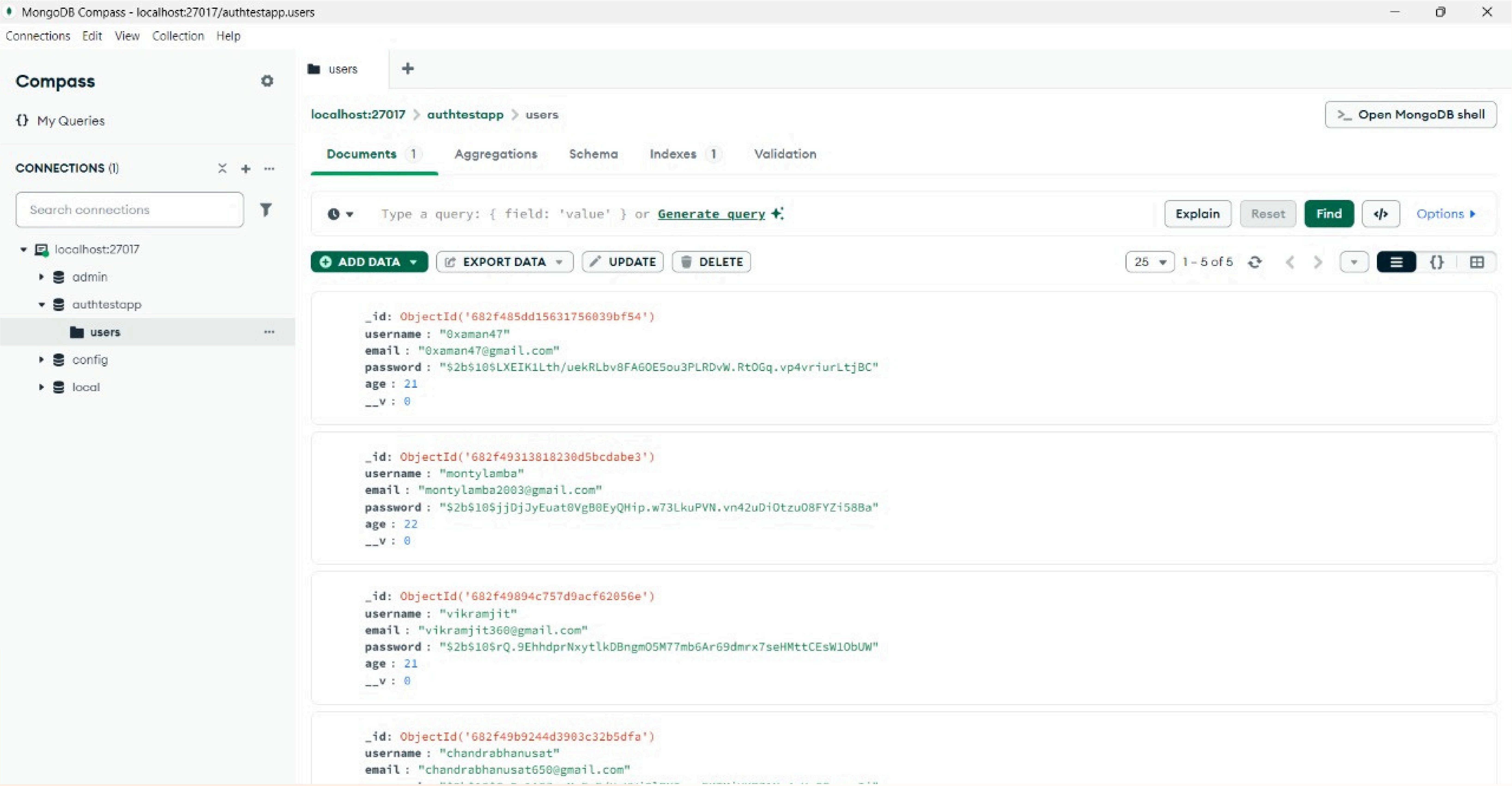

localhost:8080
WebRTC Chat Room
Login
Username
sahilwep
Password
Login
Don't have an account? Sign up
© 2025 WebRTC Chat Room. All rights reserved.

Users Intractions





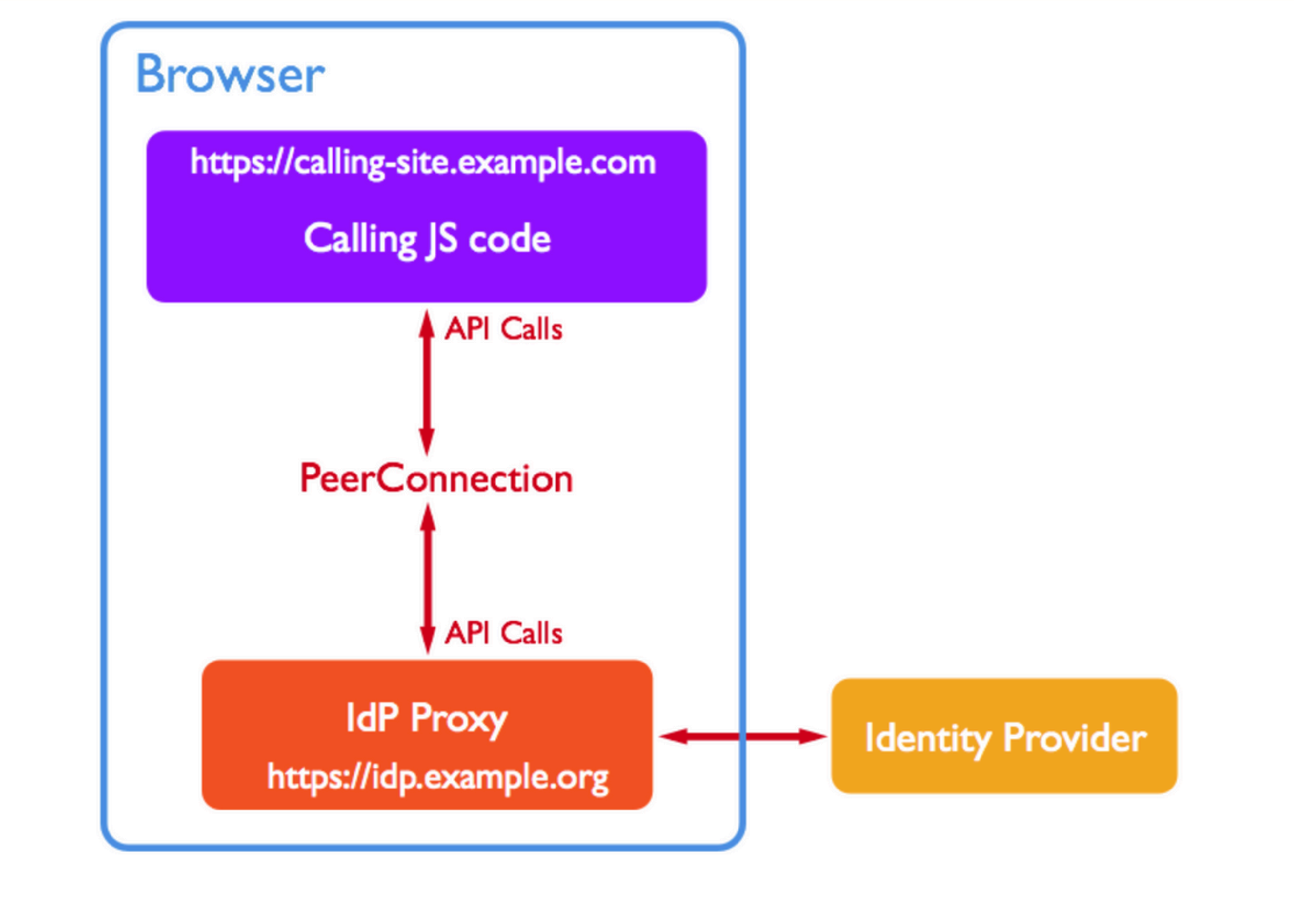
DataBase Connectivity:



AES Encryption Enables

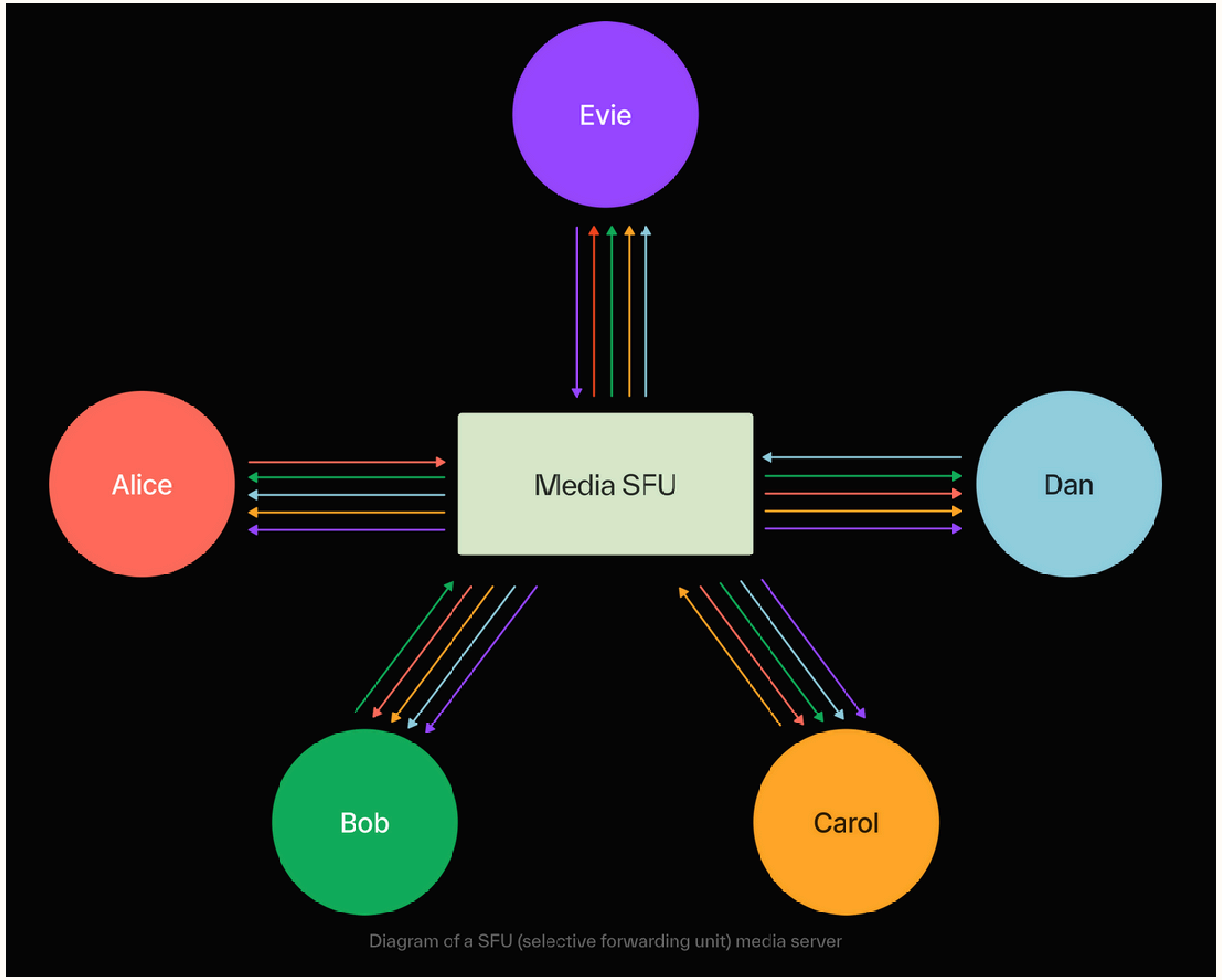

```
[{
  "_id": {
    "$oid": "682f485dd15631756039bf54"
  },
  "username": "0xaman47",
  "email": "0xaman47@gmail.com",
  "password": "$2b$10$LXEIK1Lth/uekRLbv8FA6OE5ou3PLRDvW.RtOGq.vp4vriurLtjBC",
  "age": 21,
  "__v": 0
},
{
  "_id": {
    "$oid": "682f49313818230d5bcdabe3"
  },
  "username": "montylamba",
  "email": "montylamba2003@gmail.com",
  "password": "$2b$10$jjDjJyEuat0VgB0EyQHip.w73LkuPVN.vn42uDiOtzuO8FYZi58Ba",
  "age": 22,
  "__v": 0
},
{
  "_id": {
    "$oid": "682f49894c757d9acf62056e"
  },
  "username": "vikramjit",
  "email": "vikramjit360@gmail.com",
  "password": "$2b$10$rQ.9EhhdprNxytlkDBngmO5M77mb6Ar69dmrx7seHMttCEsW1ObUW",
  "age": 21,
  "__v": 0
},
{
```

Secure: Rely on Browser security - SSL/TLS

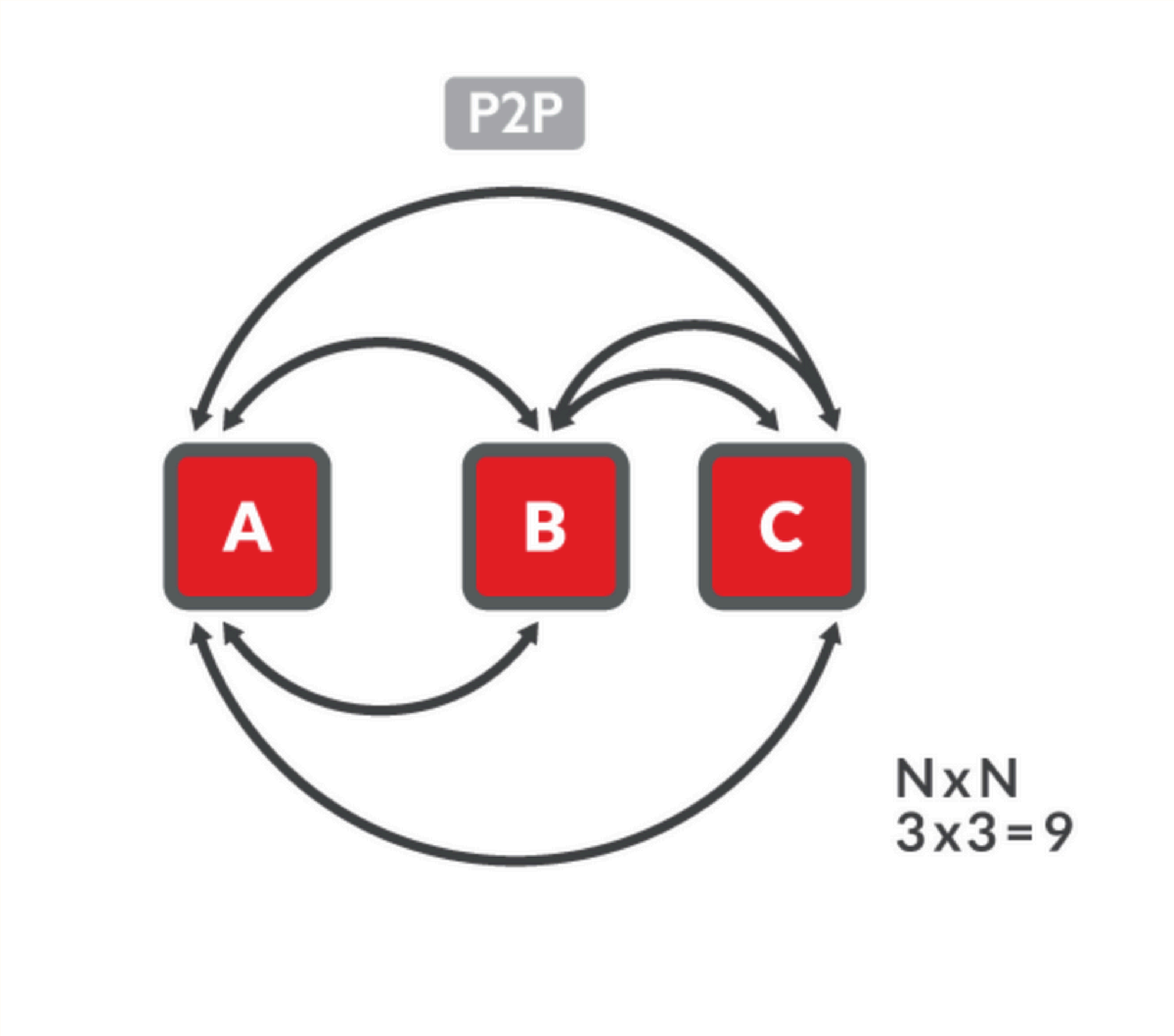


The browser's job is to enable access to the internet, while providing adequate security protections to the user. The security requirements of WebRTC are built directly upon this requirement; the browser is the portal through which the user accesses all WebRTC applications and content.

Scalable: Switch b/w p2p to SFU



SFU



P2P

Understanding Networking in WebRTC

- WebRTC uses Peer-to-Peer networking for real-time media exchange
- ICE (Interactive Connectivity Establishment) chooses the best connection path.
- STUN servers provide public IP addresses when devices are behind NAT.
- TURN servers act as relays when direct connections fail.
- NAT (Network Address Translation) and firewalls can block P2P traffic.
- WebSocket signaling is used initially to exchange metadata (SDP, ICE candidates).

Leveraging AWS for Scalable and Secure Deployment

- EC2 instances hosted the signaling server and TURN server.
- Route 53 mapped our custom domain to the EC2 server.
- AWS Certificate Manager (ACM) issued SSL certificates to enable HTTPS (required for WebRTC)
- S3 stored static frontend files (HTML, CSS, JS).
- CloudFront (CDN) delivered the frontend globally with low latency.
- AWS ensured our platform was secure, scalable, and highly available.

Tech Stack

Frontend: HTML, CSS (Bulma framework), JavaScript

Backend: Golang (Pion framework), MongoDB

Cloud Solution: AWS-EC2, VPC, S3

Future Scope

While the current system performs effectively, there is still room for improvement. Future work may include:

1) Integrating Artificial Intelligence to enable smart background noise suppression and facial recognition.

2) Incorporating edge computing to further reduce latency and improve speed.

3) Expanding compatibility with IoT devices for real-time video communication in smart environments.

4) Supporting Web3 protocols for decentralized communication. e. Enhancing the user interface for better accessibility and ease of use.

Conclusion

- This Open-Source project successfully demonstrates the design and implementation of a secure and scalable communication platform using WebRTC.
- Through experimental analysis, the system proved to offer low-latency video and audio communication, efficient peer-to-peer data sharing, and support for multiple concurrent users.
- The system also addressed security concerns by encrypting all media streams and data channels and requiring browser permissions for device access
- Additionally, the Go-based backend offered lightweight, high performance signaling services, making the platform suitable for real-time applications in healthcare, education, and remote collaboration.

Plagiarism report for research paper

Secure & Scalable WebRTC-Based Communication Platform

ORIGINALITY REPORT

15%	7%	13%	3%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	Rushali Deshmukh, Nayan Nand, Aditya Pawar, Devendra Wagh, Amol Kudale. "Video Conferencing using WebRTC", 2023 International Conference on Sustainable Computing and Data Communication Systems (ICSCDS), 2023 Publication	8%
2	research.ijcaonline.org Internet Source	1%
3	Submitted to Mar Athanasius College of Engineering Student Paper	1%
4	pure.tue.nl Internet Source	1%
5	github.com Internet Source	<1%
6	Submitted to Jose Rizal University Student Paper	<1%
7	ijariie.com Internet Source	<1%
8	"IEEE MI-STA2023 Conference Proceeding", 2023 IEEE 3rd International Maghreb Meeting	<1%

Abstract submission

Gmail Chandra Bhanu Satapathy <chandrabhanusat650@gmail.com>

International Conference on Artificial Intelligence and Sustainable Innovation - 2025 : Submission (1328) has been edited.

1 message

Microsoft CMT <noreply@msr-cmt.org> 22 April 2025 at 22:32

To: Chandrabhanusat650@gmail.com

Hello,

The following submission has been edited.

Track Name: Track 12: Session on Recent Advancements and challenges in Futuristic Technologies

Paper ID: 1328

Paper Title: Secure & Scalable WebRTC-Based Communication Platform

Abstract:
A strong open-source tool, WebRTC (Web Real-Time Communication) enables real-time transfer of video, audio, and data sharing via web browsers—without any further plugins or programs. Emphasizing its use in industries such as healthcare, online education, video conferencing, gaming, and remote collaboration, this paper looks at the architecture, functionality, and real-world uses of WebRTC. It also emphasizes WebRTC's benefits in low-latency communication, bandwidth optimization, scalability, and security via end-to-end encryption and strong protocols by comparing WebRTC-based solutions with traditional centralized systems. Signalling systems, NAT traversal techniques using STUN/TURN servers, and scalable server solutions like SFU and MCU are also discussed in the study. The work also looks at WebRTC's future possibilities in growing areas such edge computing, artificial intelligence, Web3, 5G, and IoT. This evaluation emphasizes the adaptability, low cost, and great performance of WebRTC in designing safe and flexible communication systems for both personal and business use.

Created on: Tue, 22 Apr 2025 07:15:00 GMT

Last Modified: Tue, 22 Apr 2025 17:02:45 GMT

Authors:
- komalkhanna.092@gmail.com (Primary)
- Chandrabhanusat650@gmail.com

Secondary Subject Areas: Not Entered

Submission Files:

References

- https://go.dev/doc/
- https://github.com/pion/webrtc
- https://developer.mozilla.org/en-US/docs/Web/HTML
- https://developer.mozilla.org/en-US/docs/Web/CSS
- https://developer.mozilla.org/en-US/docs/Web/JavaScript
- https://docs.aws.amazon.com/
- https://bulma.io/documentation/
- https://webrtc.org/
- https://datatracker.ietf.org/doc/html/rfc8825
- https://datatracker.ietf.org/doc/html/rfc8835

Thank you