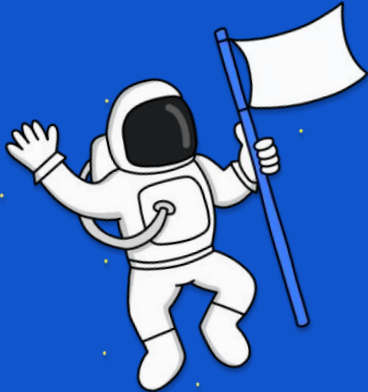


# Introduction to Real Time Web Applications

**FOSSASIA** - 20th March 2021

7.30pm to 9.00pm



# Who are we



**Low En**  
(@lowenhere)

SUTD ISTD Senior

Does software tings



**Daryll Wong**  
(@daryllman)

SUTD ISTD Senior

Does tings

**Logistics**

# Setup

Make sure you have the following:

- Git
- Node.js, along with a package manager (npm or yarn)
- A text editor / JavaScript IDE that you are comfortable with

# Course Resources

Course Webpage (along with notes)

<https://realtime-apps-iap.github.io>

Github Repository

<https://github.com/realtime-apps-iap>

This slide deck

<https://realtime-apps-iap.github.io/docs/slides/fossasia>

# Other Resources / References

Here are a few resources that have been heavily referenced in the development of the workshop, mostly because they're really comprehensive and all around great:

*Mozilla Developer Network (MDN)*

<https://developer.mozilla.org/>

*fireship.io YouTube channel*

<https://www.youtube.com/channel/UCsBjURrPoezykLs9EqgamOA>

# Agenda

- Workshop Introduction
- Introduction to WebSockets
- *Tutorial: A Simple WebSocket Server (and Client)*
- *Tutorial: A Group Chat Application*
- Fantastic Browser APIs and how to find (and use!) them
- Introduction to WebRTC
- *Tutorial: A Video Chat Application*
- Q&A

# Introduction



# What is a Real-Time Web Application?

*The real-time web is a network web using technologies and practices that enable users to receive information as soon as it is published by its authors, rather than requiring that they or their software check a source periodically for updates.*

-Wikipedia, Real-time web

*... the more vague use of the term real-time on the web, where it describes servers pushing data to clients and stream processings without hard response time constraints*

-Page 298, Designing Data-Intensive Systems, Martin Kleppmann

# Introduction to WebSockets

# WebSockets: An Internet Protocol

*A protocol defines the format and the order of messages exchanged between two or more communicating entities, as well as the actions taken on the transmission and/or receipt of a message or other event.*

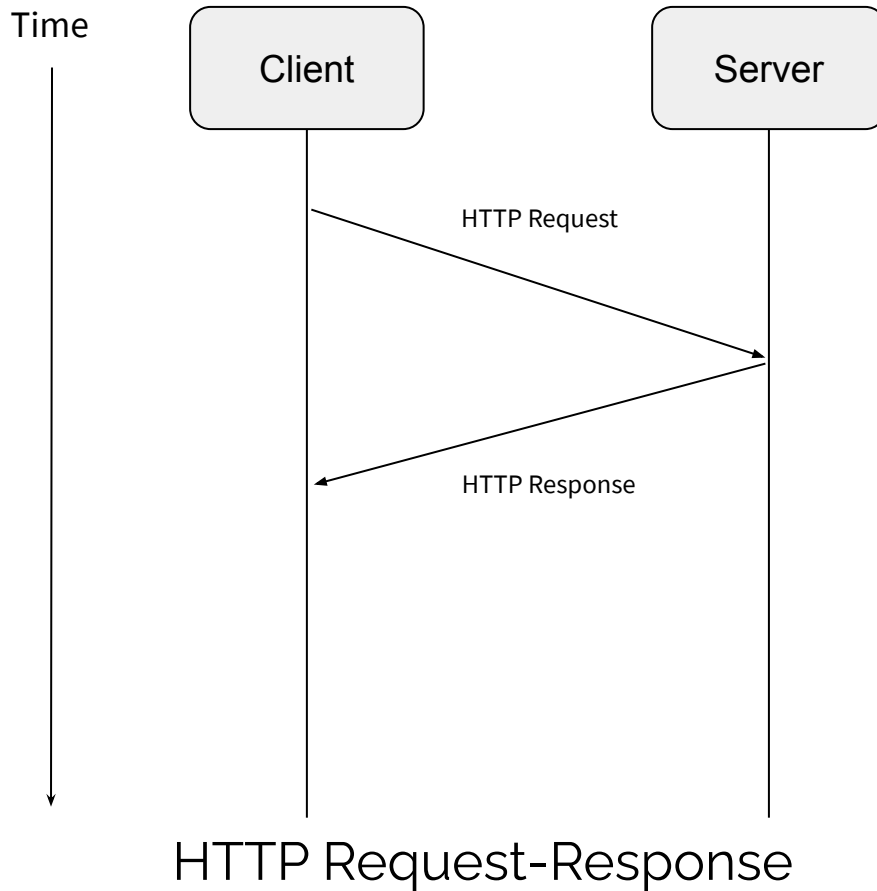
-Page 37, Computer Networking - A Top-Down Approach (7th Ed), James Kurose & Keith Ross

# WebSockets and HTTP

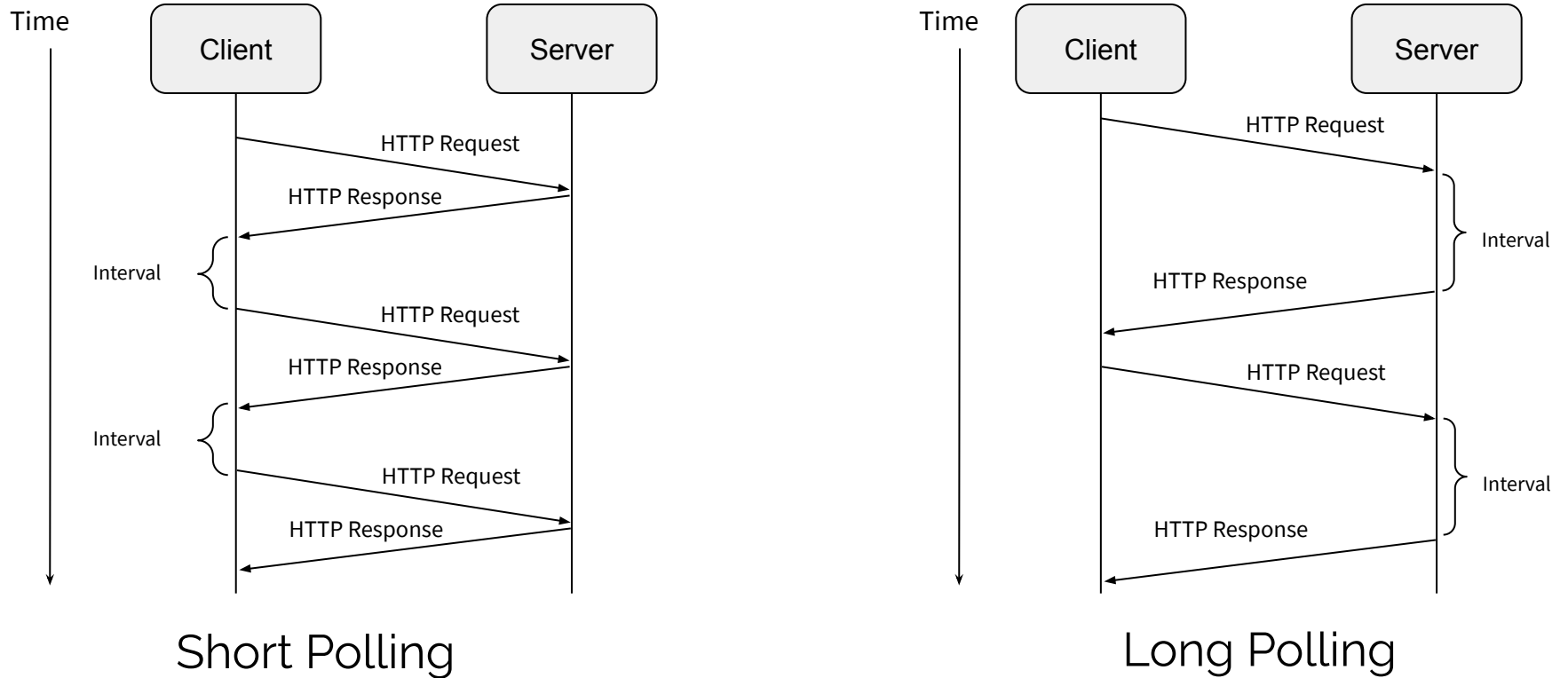
Both WebSockets and HTTP are client-server protocols.

HTTP is a request-response protocol - the client sends a request message to the server, the server returns a response message.

*Question - How do we get real-time behavior using a request-response protocol? (like HTTP)*



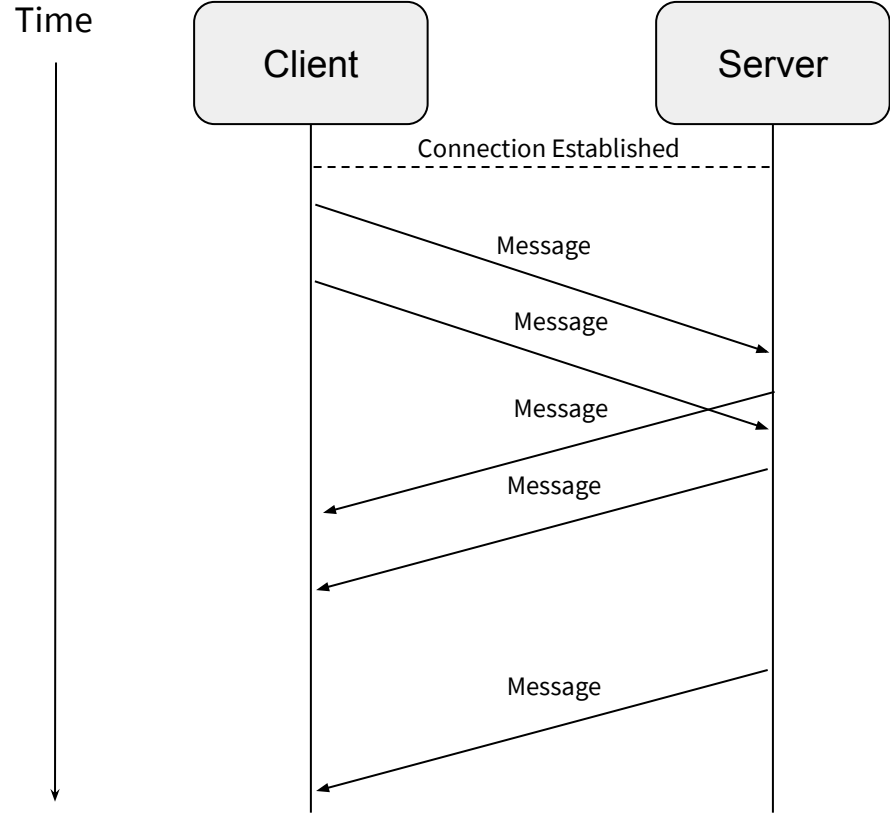
# Some hacks: Short and Long Polling



# WebSockets

## WebSocket Connections:

- Full-duplex connection, meaning that the client and server can both send data along the connection at the same time.
- Data is sent as discrete messages. Message data has to either be in a string (USVString), or some kind of binary data type (ArrayBuffer, Blob, ArrayBufferView).



WebSocket Connection

# Tutorial: Simple WebSocket Server (& Client)

<https://github.com/realtime-apps-iap/simple-websocket-server>

# Tutorial: Group Chat Application

<https://github.com/realtime-apps-iap/chat-app-starter>



# Fantastic Browser APIs and how to find (and use!) them

<https://realtime-apps-iap.github.io/docs/browserapi/main>

# Introduction to WebRTC

# WebRTC

Web Real Time Communications (WebRTC) is an open-source project that provides web browsers with with real-time communication via simple APIs.

It is a HTML5 specification you can use to add real time media communications directly between browsers

*(a robust set of APIs with versatile usage - mainly for video/audio call related applications\*)*

# What can you do with WebRTC

- Realtime Video Call applications - GoogleMeet, GoogleHangouts, Discord, etc.
- Multiplayer games - Simple FPS, WebVR, etc.

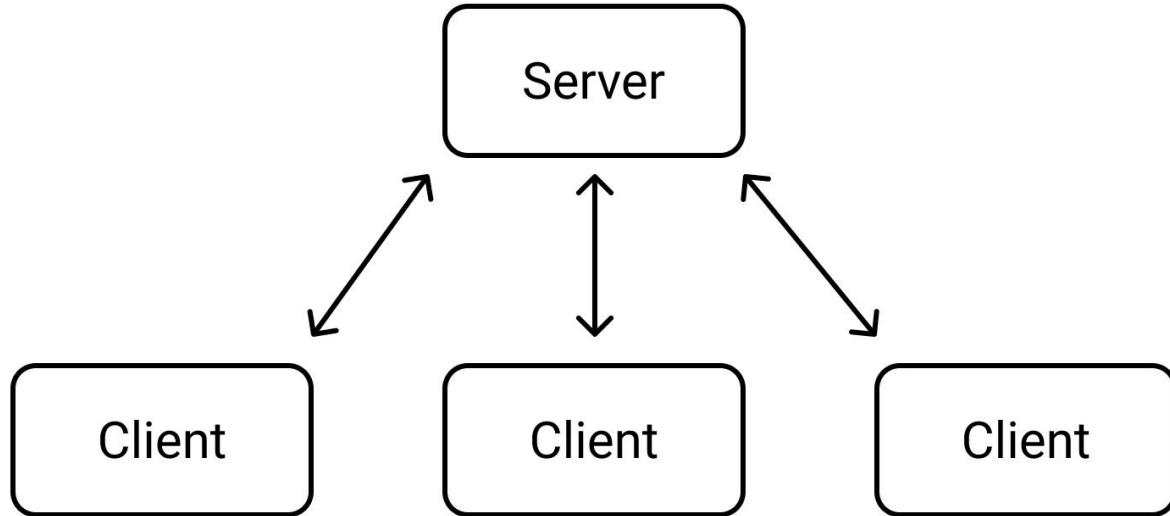
➤ **Real-Time, Audio or Video applications**

# WebSockets vs WebRTC

	<b>WebSockets</b>	<b>WebRTC</b>
Definition	WebSocket is a protocol that allows for a persistent TCP connection between server and client so they can exchange data at any time.	WebRTC provides browsers and mobile apps with Real-Time communications capabilities via simple APIs
Latency	Low Latency	Near Real-time
Data Transfer	TCP	UDP
Applications	Chat rooms, Location-based Apps, File Transfer, etc	Streaming, Video calls, Multiplayer games, etc

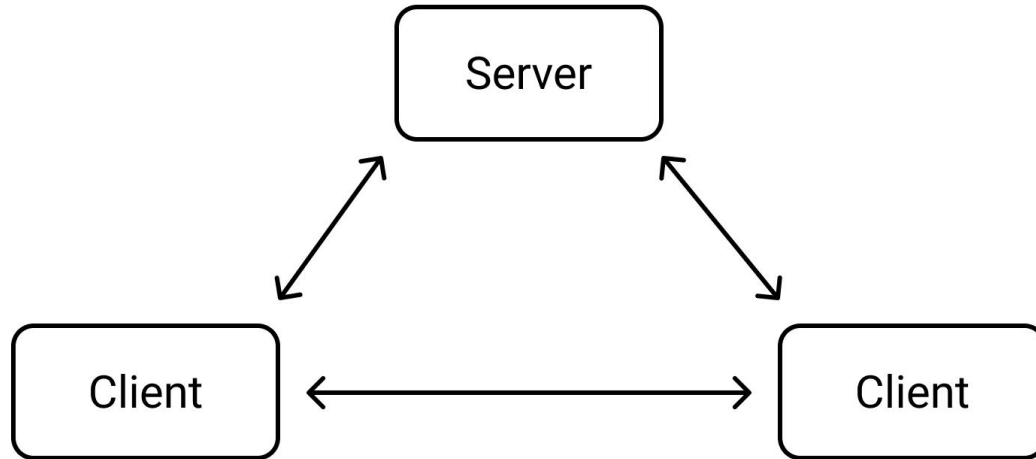
# WebSockets vs WebRTC

## WebSockets



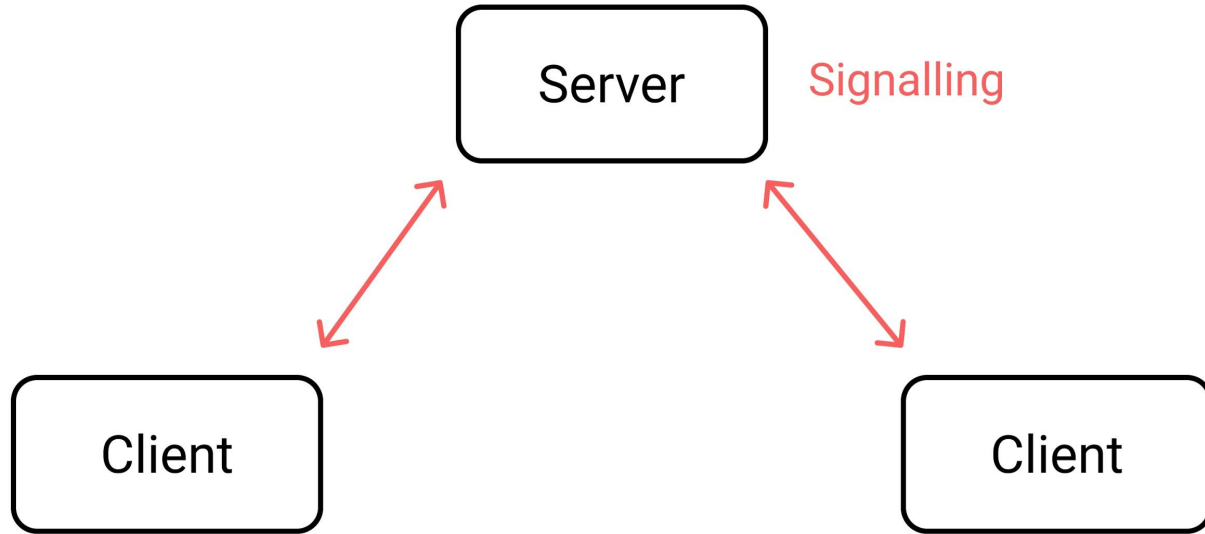
# WebSockets vs WebRTC

## WebRTC (overview)



# WebRTC

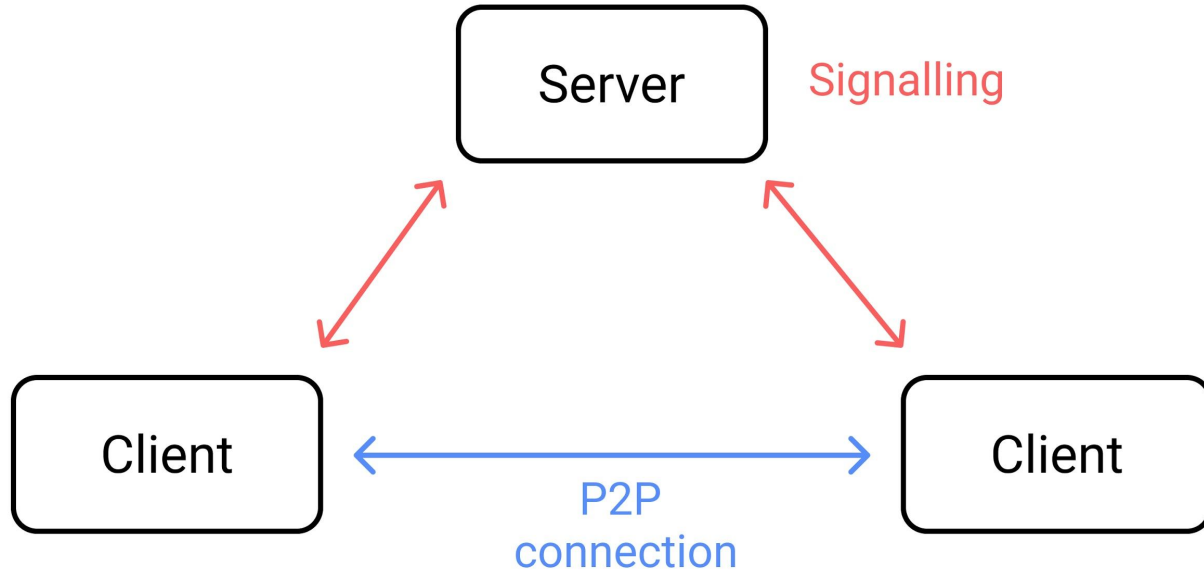
'Step 1'





# WebRTC

'Step 2'



# WebRTC

## STUN vs TURN

### 1. STUN (Session Traversal Utilities for NAT)

- STUN server identifies the public IP addresses for the client browsers to establish a Peer-to-Peer connection and exchange data

### 2. TURN (Traversal Using Relays around NAT)

- If STUN server is unable to get the IP addresses, TURN server is used as a fallback
- TURN server to act as a relay server
- A lot of data traffic through the TURN server

# Other components in WebRTC...

- Signal processing to remove simple ambient noise from audio/video feed
- Codec handling to compress & decompress data
- Routing from one peer to another through firewalls, NATs, Relays with Interactive Connectivity Establishment (ICE)
- Encrypting user data before transmitting across connections
- Managing Bandwidth to user what each peer has to give
- Etc....

# Challenges in WebRTC

- WebRTC uses UDP
  - Not a Reliable Data Transfer protocol, if this is important in the application
- No standard signaling protocol
- Not fully compatible with all browsers
  - Fine with modern versions of Chrome, Firefox
  - May have issues with Edge, Safari, older browser versions - extra plugins required

# Why WebRTC

- Great for services that requires it to be Real-Time & Fast
- No need for extra 3rd party apps
  - No need to install additional software
- Embedded in web technologies
  - Client & server can be run directly on the web browser
- Secure
  - Runs inside browser without creating new processes
  - Less chance for spyware, malware, etc
  - End-to-end Encryption

# Tutorial: Video Call Application

<https://github.com/realtime-apps-iap/video-call-app>

**A word from our (location) sponsor**

# Skills & Training Advisory Services

Skills & Training Advisory (STA) is a service provided by SkillsFuture Singapore to guide Singaporeans and Permanent Residents identify skills and training needs to support your growth at various stages of your careers, and also recommends suitable courses and programmes that caters to your needs.



<https://go.gov.sg/staoutreach>

## How Can You Benefit?



Receive personalised advisory & guidance



Discover your career interest and goals



Receive personalised advisory & guidance



Receive suitable courses recommendations



Learn about the various assistance schemes available for them



Equip yourself with the tools & resources to support your career goals



SKILLS *future* SG



**YOUR FEEDBACK IS  
VALUED**



**Share your experience  
and feedback with us!**

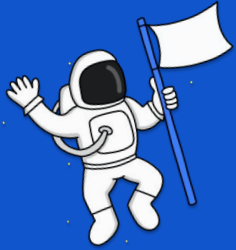


**Find out more about the  
wide range of training  
courses and resources at  
MySkillsFuture Portal**



[www.myskillsfuture.gov.sg](http://www.myskillsfuture.gov.sg)

**Scan the QR code for more  
information**



# Thank You!

We'd love to hear  
your feedback!



<https://forms.gle/y2Hpin5GF6Qv8pC49>