WebRTC School

Learn and Qualify

WebRTC School Qualified Developer (WSQD™) program

Overview

The WebRTC School™ is 'the' place to learn all about WebRTC, also known as Web Real-Time-Communications. There is so much information on the internet about WebRTC with a lot of it being hard to read, poorly presented and also lacking in detail, making it difficult for people to learn about this most important specification. So the WebRTC School™ with its lively, clear and fully animated eLearning program has become the best place to enroll to learn about WebRTC.

Who would benefit from the WebRTC School Qualified Developer (WSQD™) program?

People who looking to understand WebRTC but with a special focus on learning how to build WebRTC applications.

IMPORTANT!!! Pre-requisite knowledge

Before you purchase this course please note that it assumes knowledge of the following:

- Programming 'in general'
- JavaScript
- Basic HTML/web programming

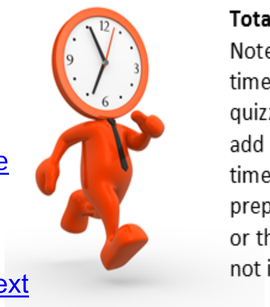
If you need information on JavaScript and HTML, we recommend http://w3schools.com. If you are a complete beginner in JavaScript, try http://codecademy.com. We have also found the book, "JavaScript: The Good Parts" a great source for understanding how to write JavaScript code in a sane way.

The Basic Real Code Walkthrough module contains server code implemented using Node.js. http://nodejs.org has both the implementation and documentation, but we have also found the Node bundle at https://leanpub.com/b/node a good way to get started.

What's in the WebRTC School Qualified Developer (WSQD™) program?

Once you've enrolled, you'll see a number of modules. You can work through the modules in order (the recommended route) or simply choose the ones you are most interested in.

1. Introduction to the course
2. Introduction to WebRTC
3. WebRTC API Preview
4. Media Flows in WebRTC
5. IETF Protocols
6. WebRTC Media Handling APIs
7. WebRTC Media Transmission APIs
8. Simple WebRTC Pseudo code example
9. The Signaling Channel
10. Basic 'Real Code' walkthrough
11. Security and Privacy
12. The 'Status' of WebRTC and What's Next
13. Use Cases
14. Interoperability and Portability



Total running time: 5hrs+15 mins.

Note: some people may take more time working with code (and quizzes) than others and this will add a considerable amount to the time you spend here. Time preparing for the certification test or the time to take the test itself is not included in this calculation

Become a 'WebRTC School Qualified Developer or WSQD™

You will be able to gain access to the test *separately* or as part of the main training package – check license 'purchase' options carefully when available.

NOTE: An access license for any training course and certification test is for 12 months from the date of purchase and if a test is purchased it must be taken before the license period expires.

Introduction to the course

Module times

- Running time = 4 minutes
- Quizzes = 0 minutes
- Total = 4 minutes

Topics Include:

- ‘Goals’ of the course
- Topics covered and the ‘approach’
- Be a WebRTC ‘Chameleon’

Introduction to WebRTC

Module times

- Running time = 14 minutes
- Quizzes = 2 minutes
- Total = 16 minutes

Topics Include:

- *Real-Time* Communication on the Internet
- WebRTC is “Skype™ in the browser”
- What's New?
- A Short History of WebRTC
- WebRTC Support of Multiple Media
- WebRTC Triangle
- WebRTC Trapezoid
- WebRTC and SIP
- WebRTC and Jingle
- WebRTC and PSTN

WebRTC API Preview

Module times

- Running time = 15 minutes
- Quizzes = 0 minutes
- Total = 15 minutes

Topics Include:

- WebRTC APIs ‘flowchart’
- Obtain Local Media
- Set Up Peer Connections
- Attach Media or Data
- Exchange Offer / Answer
    - createOffer()
    - createAnswer()
    - setLocalDescription()
    - setRemoteDescription()
- API Flow

Media Flows in WebRTC

Module times

- Running time = 9 minutes
- Quizzes = 2 minutes
- Total = 11 minutes

Topics Include:

- Media Flows in WebRTC
- Media without WebRTC
- Peer-to-Peer Media with WebRTC
- NAT Complicates Peer-to-Peer Media
- What is a NAT?
- NAT Example
- NATs and Applications
- Peer-to-Peer Media ‘through’ NAT
- ICE Connectivity Checks
- P2P Media Can Stay Local to NAT
- ICE Servers
- Browser Queries STUN Server
- TURN Server Can Relay Media
- NAT and IPv6

IETF Protocols

Module times

- Running time = 17 minutes
- Quizzes = 2 minutes
- Total = 19 minutes

Topics Include:

- IETF Protocols
- WebRTC: A Joint Standards Effort
- IETF Standards
- The WebRTC Protocol Stack
- WebRTC Protocols
- Internet Protocols
- Transmission Control Protocol
- User Datagram Protocol
- Transport Layer Security
- Datagram Transport Layer Security
- Stream Control Transport Protocol
- Hypertext Transport Protocol
- The WebSocket Protocol
- Secure Real-Time Transport Protocol
- Session Description Protocol
- Session Traversal Utilities for NAT
- Traversal Using Relays around NAT
- Interactive Connectivity Establishment
- What about SIP?

WebRTC Media Handling APIs

Module times

- Running time = 38 minutes
- Quizzes = 2 minutes
- Total = 40 minutes

Topics Include:

- W3C WebRTC standards work
- Standards process and conformance
- Two API modules
- Local Media Handling Example
- Sources
- Tracks
- MediaStreamTrack API
- The constraint approach
- Constraints
- States
- Capabilities
- SourceInfo
- MediaStreamTrack API
- MediaStreamTrack subclasses
- Streams
- MediaStream API
- getUserMedia()

WebRTC Media Transmission APIs

Module times

- Running time = 48 minutes
- Quizzes = 2 minutes
- Total = 50 minutes

Topics Include:

- A Peer Connection
- WebRTC Triangle ‘Review’
- A Peer Connection is
- About ICE
- WebRTC NAT Traversal
- ICE Call Flow
- ICE Gathering State Machine
- ICE Connection State Machine
- Offer/Answer
- Offer/Answer State Machine
- Media Description
- MediaStreamTrack API
- RTCPeerConnection Core API
- “Extra” APIs
- DTMF
- DTMF in WebRTC
- DTMF API
- Data Channel
- Data Channel API
- Identity API
- Statistics

Simple WebRTC Pseudo code example

Module times

- Running time = 30 minutes
- Quizzes = 2 minutes
- Total = 32 minutes

Topics Include:

- Pseudo Code
- Mobile browser code outline
- Mobile browser 'streams' example
- function getMedia()
- function createPC()
- Function handleIncomingStream()
- Function show_av(st)
- Mobile browser code outline
- function attachMedia() [1]
- function call()
- How do we get the SDP answer?
- Laptop browser stream example
- Signaling channel message is trigger
- Function prepareForIncomingCall()
- Function answer()
- Function handleIncomingStream()
- Function getMedia() [1]
- Function attachMedia()
- In real code . .

The Signaling Channel

Module times

- Running time = 13 minutes
- Quizzes = 2 minutes
- Total = 15 minutes

Topics Include:

- WebRTC signaling
- Role of Signaling
- Why Signaling is Not Standardized
- Server Chooses Signaling Protocol
- Some Signaling is Needed
- Signaling in WebRTC
    - Signaling State Machine
    - Signaling Transport Options
    - Signaling Transport Example: WebSockets
    - Signaling Transport Example: HTTP
    - Signaling Transport Example: Data Channel
    - Signaling Protocol Options
    - Signaling Example: HTTP Polling
    - Signaling Example: WebSockets
- WebSocket code outline
- Standardized protocols
- Open Source JavaScript SIP Stacks
- Open Source JavaScript XMPP
- Comparison of Approaches

Basic ‘Real code’ example

Module times

- Running time = 53 minutes
- Quizzes = 3 minutes
- Total = 56 minutes

Topics Include:

- Overview
- Structure of module
- Web server
- Node.js
- Index and Server
- Signaling channel
- Signaling channel interface
- Server Signaling
- Client Signaling
- Web application
- What’s Next?

Security and Privacy

Module times

- Running time = 35 minutes
- Quizzes = 2 minutes
- Total = 37 minutes

Topics Include:

- What are Security and Privacy?
- Is WebRTC Secure and Private?
- Web Security and Privacy Model
- Browser/Web Security Model
- Browser/Web Privacy Model
- How WebRTC Changes Browser Privacy
- Browser Prompts for Permission
- How WebRTC Changes Browser Security
- New Attacks
- WebRTC API Attacks
- Signaling Channel Attacks
- Security of WebRTC Media Sessions
- Website Identity
- Browser User Identity
- Secure Real Time Protocol
    - Secure Profile of RTP (SRTP)
    - SRTP Encryption and Authentication
    - SRTP Key Management
    - Key Exchange in Signaling
    - Generating a Key in Media Path*
    - SDP Security Descriptions
    - Generating a Key in Media Path
    - SRTP Call Flow
- DTLS-SRTP Key Agreement
- DTLS Client/Server
- Authenticating a Key Agreement
- Authenticating a Fingerprint
- Identity Proxy in WebRTC Triangle
- Single Identity Proxy with Triangle
- Identity Proxy in WebRTC Trapezoid
- Identity Services
- Communication Consent
- ICE Communication Consent
- Privacy in WebRTC
- Identity Privacy
- IP Address Privacy
- Browser Fingerprinting
- Media Privacy
- WebRTC and the Enterprise
- WebRTC Security Summary

Status of WebRTC and ‘What’s next’

Module times

- Running time = 5 minutes
- Quizzes = 0 minutes
- Total = 5 minutes

Topics Include:

- Status of WebRTC APIs
- Status of WebRTC Protocols
- Status of Browser support of WebRTC
- Support of Signaling Channel
- Support in Mobile
- Opportunities
- Obstacles
- What’s next?

Use Cases

Module times

- Running time = 7 minutes
- Quizzes = 0 minutes
- Total = 7 minutes

Topics Include:

- WebRTC Use Cases and Requirements
- Web Conferencing (Multiparty Conferencing)
- Communications Client (UC and Consumer)
- Contact Centers (B2C and Agent)
- Distributed Communication (Freemium Services)
- Mobile (Voice for Smartphones)
- Single Line of Code WebRTC (Libraries)
- Control (Microphone and Cam Access)
- Gaming (In-Game Media, Chat and Data Channel)
- Overlay Network (Data Channel)

Interoperability and Portability

Module times

- Running time = 10 minutes
- Quizzes = 0 minutes
- Total = 10 minutes

Topics Include:

- Definitions
- Relative importance
- Things to watch
- Implemented APIs still prefixed
- Connecting to output is different
- No default STUN servers
- Multiple audio/video not supported
- Opus not default
- DTLS-SRTP
- Staying current
- Other resources