# Use-Cases of the new 3GPP Immersive Voice and Audio Services (IVAS) Codec and a Web Demo Implementation


Eleni Fotopoulou
*DSP Solutions GmbH & Co. KG*
Regensburg, Germany
eleni.fotopoulou@dspsolutions.de

Kacper Sagnowski
*Fraunhofer IIS*
Erlangen, Germany
kacper.sagnowski@iis.fraunhofer.de

Karin Prebeck
*Fraunhofer IIS*
Erlangen, Germany
karin.prebeck@iis.fraunhofer.de

Moumita Chakraborty
*Fraunhofer IIS*
Erlangen, Germany
moumita.chakraborty@iis.fraunhofer.de

Suhas Medicherla
*Fraunhofer IIS*
Erlangen, Germany
suhas.medicherla @iis.fraunhofer.de

Stefan Döhla
*Fraunhofer IIS*
Erlangen, Germany
stefan.doehla@iis.fraunhofer.de



***Abstract*— The newly standardized 3GPP codec for Immersive Voice and Audio Services (IVAS) is the first codec specifically tailored for immersive communication within 5G mobile networks. The IVAS codec introduces capabilities for coding and rendering of stereo, multi-channel, Ambisonics, audio objects, and the innovative metadata-assisted spatial audio format. It paves the way for new service scenarios involving interactive stereo and immersive audio communication, content sharing, and distribution. This paper describes a demonstration of three exemplary IVAS codec use-cases—multi-party conferencing, immersive calls and ad-hoc conferencing—using a tablet with headphones. Attendees can experience the impact of spatial audio in future mobile communication services firsthand, comparing the IVAS codec interactively with its mono predecessor, EVS, which is currently the state-of-the-art in mobile networks. Finally, the technical details and building blocks for the demo implementation are described.**



**Keywords—3GPP, IVAS, Audio Codec, Spatial Audio, 5G Networks, Communication Systems, Demonstration.**


## I. Introduction

### A. Overview

The Third Generation Partnership Project (3GPP) has recently completed its Release 18 ("5G-Advanced") set of telecommunications standards, introducing numerous new functionalities and features, including a groundbreaking codec for Immersive Voice and Audio Services (IVAS). The standardization of the IVAS codec marks a significant step in 3GPP's ongoing efforts to remain at the forefront of voice and audio service innovation while ensuring highly efficient service delivery. This initiative is driven by the growing demand for enhanced quality of experience, aligned with the broader trend towards sound-based interactions as outlined in [1].

Despite the high quality of state-of-the-art 3GPP voice services such as the 3GPP Enhanced Voice Services (EVS) codec ([2], [3]) the limitation of only offering monophonic audio experiences remains. For users to be able to fully immerse themselves in an audio scene, which is essential for true "being there" experiences, spatial audio is necessary. Consumers are already familiar with immersive audio through other applications, like movie theatres, surround sound home theatre and music systems, or applications on their mobile devices. However, spatial audio is still not available for mobile communications. 3GPP has addressed this gap by developing the new IVAS codec standard.

The new codec comes with numerous capabilities and features for coding and rendering. However, the potential applications of IVAS are not always clear to the community, including service providers and device manufacturers. Consequently, there is a pressing need to implement example use cases that demonstrate how to leverage these new capabilities, which are unprecedented in traditional 3GPP communication codecs.

### B. Envisioned Use-Cases

The transmission of spatial and immersive audio with the IVAS codec enables new communication scenarios [4], such as:

- Multi-party Conferencing: In a typical conferencing situation, multiple participants' voices can be transmitted as individual streams and spatially rendered on the receiving device to match the concurrently transmitted video scene. This also allows for the separation and manipulation of each participants' voice individually. An intermediate call server can also merge multiple participants from different locations into a virtual immersive scene.
- Immersive Calls (experience sharing): Allows participants to additionally capture immersive scenes and convey them to one another. The full immersive audio experience of, e.g., an event or outdoor environment can be shared.
- Ad-hoc Conferencing: By placing a device on a conferencing table, a realistic acoustic image of the surrounding participants can be captured and recreated at one or multiple receivers. This immersive rendering makes it easier to distinguish between speakers' voices and separate them from ambient sounds.

These use cases are just a few examples of the potential applications of the IVAS codec. IVAS can also be utilized for applications beyond traditional communication services, such as live and non-live streaming of user-generated immersive

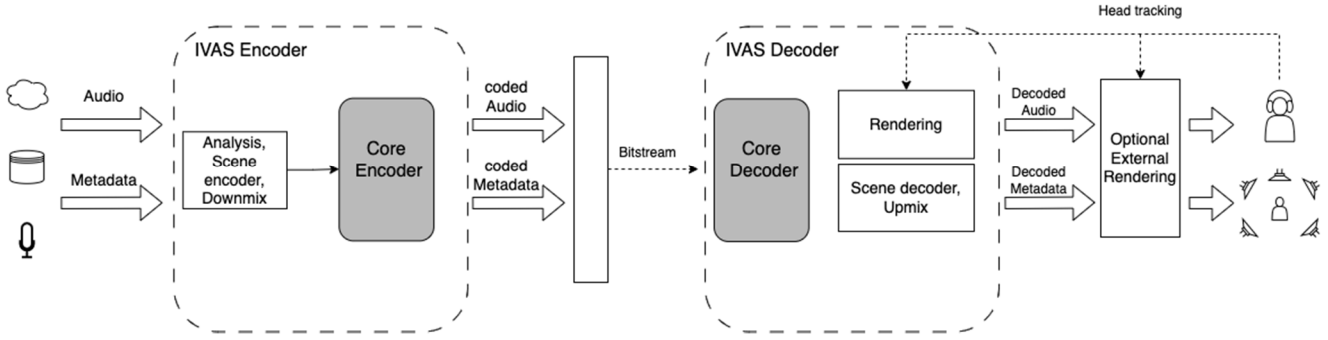


Fig. 1. Block diagram of the IVAS codec architecture

and Extended Reality (XR) content, as well as advanced XR and metaverse applications [5].

### C. Beyond Typical Communication Scenarios

The emerging fields of the Internet of Sounds (IoS) [1] and the Internet of Musical Things (IoMuT) [6] describe several use-cases such as immersive audio experiences in smart environments (e.g. emergency scenes or telemedicine), interactive audio installations, and music-based augmented reality applications [7], [8]. On the other hand, advancements in 5G networks including new features such as slicing and multi-access edge computing (MEC) can enable application scenarios envisioned in these fields as described in [9], [10], [11].

The new IVAS codec could play an essential role for the audio acquisition and reproduction offering efficient coding schemes and various rendering capabilities, including head-tracking, orientation tracking and split-rendering for lightweight less capable devices [12].

In this paper, the implementation of a web demo of the three use-cases outlined above is presented. In section II a brief overview of the key features of the new codec is given. In section III the demo concept is described, giving an overview of the specific scenarios used and their interface. In section IV technical details of the demo implementation are described, followed by conclusions in section V.

## II. Technical Overview of Codec Features

The IVAS codec is an extension of the 3GPP EVS mono codec to stereo and immersive coding. Its main attributes are coding and rendering of immersive signals, broad range of bitrates spanning from low-bitrate efficient coding to excellent quality high-bitrate coding, low delay suitable for communication applications, and moderate implementation complexity/memory requirements. A block-diagram of the IVAS codec architecture is given in Fig. 1.

A detailed technical description of the IVAS codec can be found in [4], [12]. The basic capabilities and features are summarized below.

### A. Supported Formats

The input formats supported by the IVAS codec are the following:

- Mono Bit-Exact EVS compatibility: Complete support for mono speech/audio signal input.
- Stereo and Binaural Audio: Enhanced support for immersive audio experiences.
- Spatial Audio Formats:
  - Multi-channel audio configurations: 5.1, 5.1.2, 5.1.4, 7.1, 7.1.4
  - Scene-based audio: 2D (planar) or 3D Ambisonics [13], order 1 - 3
  - Metadata-Assisted Spatial Audio (MASA): A parametric format for direct spatial audio capture from smartphones [14]
  - Object-based audio.
- Combined Immersive Audio Formats:
  - Object-based audio with scene-based audio (OSBA)
  - Object-based audio with metadata-assisted spatial audio (OMASA)

### B. Codec Features

The codec is optimized for services over 5G mobile networks and implementations on 5G devices, with its main attributes shown in Table I [4].

### C. Decoder and Rendering Capabilities

The codec comes with integrated rendering algorithms that are tightly coupled to the scene decoder/upmixer. Next to providing the "pass-through" decoded audio signal and metadata, the IVAS decoder offers:

- Rendering functionality for loudspeakers playback and binaural rendering for headphone reproduction

TABLE I. IVAS KEY FEATURES IN STEREO AND IMMERSIVE FORMATS.

| | |
|---|---|
| Sampling rates | 16, 32, 48 kHz |
| Audio bandwidths | WB (20 – 8,000 Hz),<br>SWB (20 – 16,000 Hz),<br>FB (20 – 20,000 Hz) |
| Bitrates | 13.2, 16.4, 24.4, 32, 48, 64, 80, 128, 160, 192, 256, 384, 512 kbps |
| Frame length | 20 ms (encoder/decoder)<br>5/10/20 ms (renderer) |
| Algorithmic delay | 32 to 38 ms<br>(incl. rendering delay) |
| Audio formats | mono, stereo, scene-based audio, object-based audio, multichannel-based audio, MASA |
| Output configurations | loudspeakers (mono, stereo, 5.1, 5.1+2, 5.1+4, 7.1, 7.1+4), Ambisonics (1 - 3 order), binaural, pass-through |


Fig. 2. Menu interface of the demo

- Head-tracking, scene rotation, reverberation, and support for external custom Head-Related Transfer Functions (HRTFs).
- A split rendering solution for head-tracked rendering on lightweight devices.
- The option to bypass the built-in renderer to use custom renderers.

### D. *Efficiency and Resilience for Mobile Networks*

IVAS comes with additional elements essential for mobile network communications.

- Rate Efficiency: Features such as Voice Activity Detection (VAD), Discontinuous Transmission (DTX), and Comfort Noise Generation (CNG) enhance rate efficiency for stereo and immersive conversational voice transmissions.
- Error Resilience: Mechanisms for error concealment to mitigate transmission errors and packet loss.
- Jitter Buffer Management (JBM): Ensures smooth audio playback.

## III. Demo Concept

The demo utilizes a web-based app installed on a tablet equipped with the IVAS codec and high-quality noise-cancelling headphones. The primary objective is to enable a direct comparison between the IVAS codec and the 3GPP EVS codec, which is currently the mandatory codec for Voice over LTE (VoLTE) and Voice over New Radio (VoNR). Attendees can instantly switch playback between the two codecs, allowing for a comparison between current state-of-the-art monaural audio communication and the potential immersive experience provided by future IVAS-enabled services.

For encoding the demonstrated scenes, the same bitrate of 64 kbps is used for both codecs. For EVS, the immersive audio input signal is rendered to mono prior to encoding. By maintaining the same bitrate, the demo additionally highlights the efficiency of the IVAS codec. Despite encoding a multi-channel signal—16 channels in the case of 3rd order Ambisonics—IVAS achieves perceptual audio quality comparable to EVS. For the binaural playback of the IVAS output, real-time rendering of the pre-decoded IVAS output is performed using the external IVAS renderer provided with the reference source code [15]. There is no underlying 5G network emulation implemented, therefore, both conditions are coded assuming clean channel transmission, i.e. without any packet loss concealment.

The interface of the web-based application is intuitive. First the user is invited to select the use-case to be demonstrated, as shown in Fig. 2. Some use-case specific details are given below.

### A. *Multi-party Conferencing*

The example of multiple participants being captured separately as independent audio streams and that are rendered spatially from e.g. a call server in a pre-defined scene is demonstrated.

In the demonstrated scenario three people on the remote scene are moving around in the workplace coffee room having a conversation. The attendee, as the remote participant, can listen to the conversation and distinguish their positions in the room as they stand but also, as they are moving around him.

The coding of audio objects is demonstrated, enabling users to manipulate these objects by creating a customized audio scene or adjusting their volume. This also allows users to mute or enhance any of the three objects relative to the others.

The interface for the demonstration of this use case is given in Fig. 3. Apart from the basic idea of comparing to EVS, the ability to manipulate the audio objects is also given. Some visual animation helps the user understand the scene and where the talkers are positioned around him. The talkers are symbolized in the form of coloured cubes, each colour represents a different talker and the active object blinks. The user can manipulate the scene by dragging and moving the cubes around. The ability to manipulate the gain for each object is given from the slide bars with the respective color as shown in Fig. 3.

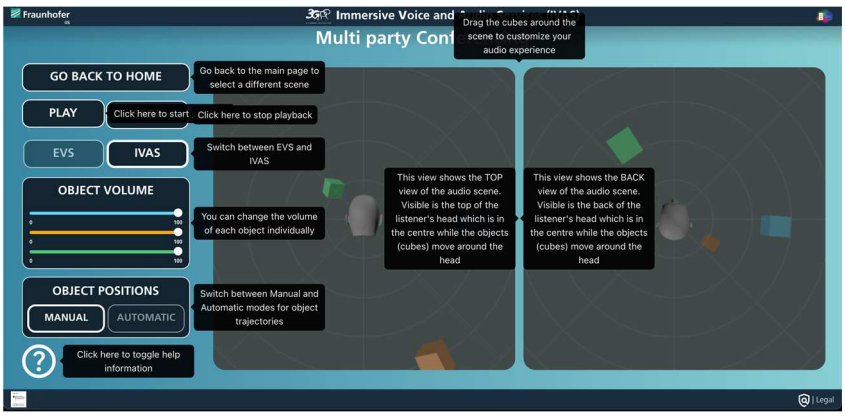


Fig. 3. Interface of the "Multi-Party Conferencing" use-case

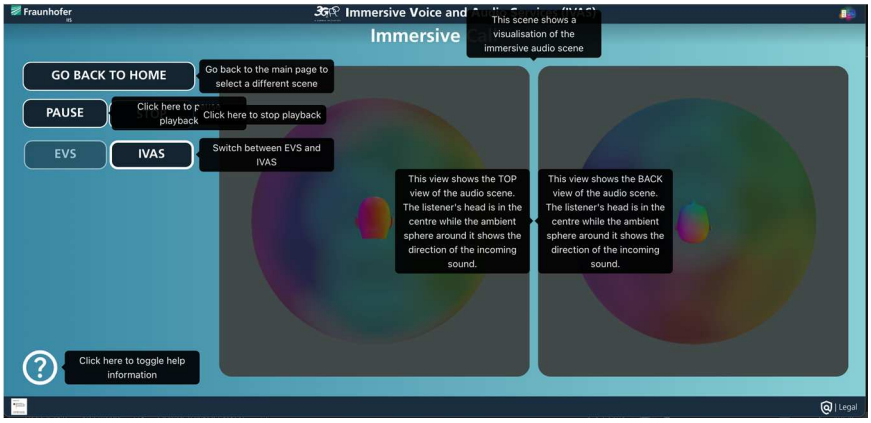


Fig. 4. Interface of the "Immersive Call" use-case

### B. *Immersive calls (experience sharing)*

This is to demonstrate the so-called "experience sharing" scenario. The participant on the one end is capturing the immersive scene in a forest with a river flowing and some live music in the background with an Ambisonics microphone, while describing on top what the person is experiencing using the handset microphone. In the demo, the attendee is the participant on the other end.

This is an example scenario where we have combined formats, namely Ambisonics together with a mono signal or an object. For the purpose of the demo, the forest scene is taken from the QoEVAVE database [16], captured with an Eigenmike microphone, combined with a separate recording of a talker describing the scene. The mono object is prerendered to the HOA3 audio scene prior to encoding.

The attendee has the ability to switch seamlessly between IVAS and EVS. In this way, one can experience directly the difference of using spatial audio to experience the scene, versus to what he or she would experience with a conventional phone call today. The interface is simple and visual animation help the user understand where the sound is coming from in the form of an acoustic sphere as can be seen in Fig. 4.

### C. *Ad-hoc Conferencing*

This demonstrates the simple use-case of an Ambisonics microphone placed on a desk in an office where people are sitting on different positions having a conversation.

The scene was captured with an Eigenmike microphone and was also selected from the QoEVAVE database [16]. This is coding of scene-based audio with Ambisonics.

The interface is the same as for the *Immersive Call,* shown in Fig. 4*,* the user has the ability to compare to EVS while the animation reproduces visually the direction of the sound relative to the "remote" user.

## IV. Demo App Implementation

The following sections give a top-down overview of key implementation aspects of the IVAS Web Demo App.

### A. *Building the Android App*

The IVAS Demo App was implemented in JavaScript as a web app and therefore, its primary target platforms are web browsers. However, due to the ubiquity of web technologies, many methods exist for embedding them in other environments, making them effectively cross-platform.

It is most convenient to show the demo as a native app on an Android tablet to make it portable. To accomplish this, the Capacitor.js [17] runtime was used—a tool designed specifically to convert JavaScript apps into native Android and iOS applications. The process of creating an Android app with Capacitor consists of two main steps. First, an Android project is generated from an existing JavaScript codebase using the command line tool provided by Capacitor. In the case of the IVAS Demo App, minor adjustments were made to the resulting project to adapt the application to the requirements of showing the demo at conventions and conferences, e.g. screen timeout was increased. The second step is to build the app using the Gradle build tool [18] , which is standard in Android app development.

### B. *Graphical User Interface*

The graphical user interface (GUI) was implemented using the React framework [19], which was chosen for its ease of use and rich feature set. Key features that facilitated the development of the app were, among others, granular updates of GUI components, built-in support for cross-page navigation and easy integration with an extensive ecosystem of third party libraries.

One of the main goals of the Demo App was to convey the ideas behind spatial audio by providing visualizations of audio scenes to the listener. This was implemented using the three.js [20] library, which takes advantage of Web Graphics Library (WebGL) support in browsers and web views to enable rendering and controlling of 3D scenes in real time. Thanks to the high-level JavaScript API that three.js provides, it was possible to easily create dynamic relationships between audio and visuals without sacrificing performance. In one direction, user interactions with the visualization (e.g. dragging 3D objects) are captured by the event system built into three.js and are used to affect the parameters used for rendering of spatial audio. In the other direction, the sound intensity of each element of the spatial audio scene is mapped in real time to properties of three.js objects, resulting in a dynamic audio activity visualization.

### C. *Web Audio API*

With sound being the focal point of the app, it was crucial to use audio processing methods that ensure consistently high performance and provide seamless audio playback in real time. In the context of web technologies, the Web Audio API (WAA) [21] offers a comprehensive and reliable framework for achieving these goals. Practical considerations of the use of WAA for spatial audio processing have been discussed in

[22] [23]. Additionally, example integrations of WAA into web applications with a focus on spatial audio have been described in [24][25].

The core functionality of WAA is provided by audio nodes, which can be connected into larger processing graphs. The API comes with a number of specialized audio node types for dedicated tasks, such as, for example, visualizer nodes, which were used to drive the animation in the 3D preview based on audio activity or gain nodes, which were used in the app for fading between EVS and IVAS streams (note that, conversely, adjustable gain for each audio object was achieved using built-in IVAS functionality).

WAA also offers the possibility to define custom audio processors that, crucially, can execute on a dedicated audio thread with real-time priority and low latency. A corresponding audio node is instantiated on the main thread for controlling the custom processor and creating connections to other nodes in the audio processing graph. This mechanism was utilised to include an IVAS renderer instance in the audio processing graph of the application. Rendering parameters that change at runtime, such as object positions or gain, were defined with audio parameter descriptors (also part of WAA) and thus automatically exposed as modifiable parameters of the corresponding audio node, allowing to affect spatial rendering based on changes in the GUI.

### *D. Integrating IVAS with web technologies*

Web browsers cannot directly execute native binaries compiled from C source code, such as the IVAS library. Compiling to Web Assembly (Wasm) [26] provides a way to run code written in multiple languages (including C and C++) at near-native speed within the browser.

For integration within the Demo App, a C++ wrapper around the IVAS C library was created to provide an object-oriented abstraction over the IVAS API. Subsequently, the wrapper library was compiled to Wasm using the Emscripten [27] compiler toolchain. In the process, JavaScript bindings were also generated so that C++ functions could be called from JavaScript. The resulting package was integrated via the JavaScript bindings into the Demo App as part of the custom audio processor.

## V. DISCUSSION AND CONCLUSION

In this paper, a web demo implementation of the 3GPP IVAS codec, which highlights its groundbreaking capabilities in bringing immersive audio experiences to mobile communications, is presented. The technical overview presents the extensive capabilities of the IVAS codec, from supporting various immersive audio formats to offering efficient, low-delay, and resilient performance suitable for mobile networks. With the presented demo implementation, users have the opportunity to experience firsthand the advancements offered by the IVAS codec and the potential of immersive audio in enhancing user experiences across three use-cases, including multi-party conferencing, immersive calls and ad-hoc conferencing. The description of the demo app's implementation provides insightful information on utilizing modern web technologies and cross-platform frameworks. It demonstrates how the IVAS API can be practically employed to integrate high-quality spatial audio coding and rendering on a mobile device.

IVAS is designed for mobile networks and immersive telephony use-cases; however, its extensive features can accommodate applications beyond typical communication scenarios, including those envisioned by the IoS community as outlined in [1]. The IVAS codec is part of 3GPP Release 18 ("5G Advanced"), and as such, large deployments on mobile devices are foreseen. IVAS is expected to be integrated into LTE/5G networks and terminals, enabling future immersive services. This would, for the first time, enable broad usage of stereo and immersive audio communications, providing new options that could be valuable for IoS applications as well. As a communication codec targeting low and medium bitrates, IVAS incurs an algorithmic delay of 32 – 38 ms, which is tolerable for communication services on mobile devices. Consequently, IVAS is suitable for applications that could tolerate a minimum end-to-end delay of 125 ms—typical for mobile device communication. Additionally, the IVAS decoder includes integrated renderer and JBM functionality, which help reduce overall end-to-end latency.

In conclusion, the IVAS codec provides a standardized framework for immersive audio in 5G mobile networks, allowing the IoS community and researchers from various disciplines to explore its numerous features and capabilities for future innovative use-cases, potentially enabling advancements in networked immersive audio experiences and user interactions.

## ACKNOWLEDGMENT

This work has partly been funded by the German Federal Ministry for Economic Affairs and Climate Action (ToHyVe, grant no.01MT22002A)

## REFERENCES

[1] L. Turchet *et al*., "The Internet of Sounds: Convergent Trends, Insights, and Future Directions," in *IEEE Internet of Things Journal*, vol. 10, no. 13, pp. 11264-11292, 1 July1, 2023,

[2] M. Dietz *et al*., "Overview of the EVS codec architecture," *2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, South Brisbane, QLD, Australia, 2015, pp. 5698-5702.

[3] 3GPP TS 26.445, "EVS Codec Detailed Algorithmic Description; 3GPP Technical Specification," 2014.

[4] M. Multrus *et al.*, "Immersive Voice and Audio Services (IVAS) Codec – The New 3GPP Standard for Immersive Communication," *157th AES Convention*, October 2024, in press.

[5] M. Sparkes, "What is a metaverse", *New Sci.*, vol. 251, no. 3348, pp. 18, Aug. 2021.

[6] L. Turchet, C. Fischione, G. Essl, D. Keller, and M. Barthet, "Internet of Musical Things: Vision and challenges," *IEEE Access*, vol. 6, pp. 61994–62017, 2018

[7] C. Rottondi, C. Chafe, C. Allocchio and A. Sarti, "An overview on networked music performance technologies", *IEEE Access*, vol. 4, pp. 8823-8843, 2016.

[8] L. Turchet, R. Hamilton and A. Çamci, "Music in extended realities", *IEEE Access*, vol. 9, pp. 15810-15832, 2021.

[9] F. Martusciello, C. Centofanti, C. Rinaldi and A. Marotta, "Edge-Enabled Spatial Audio Service: Implementation and Performance Analysis on a MEC 5G Infrastructure," *2023 4th International Symposium on the Internet of Sounds*, Pisa, Italy, 2023, pp. 1-8.

[10] L. Turchet, C. Rinaldi, C. Centofanti, L. Vignati and C. Rottondi, "5G-Enabled Internet of Musical Things Architectures for Remote Immersive Musical Practices," in *IEEE Open Journal of the Communications Society*, vol. 5, pp. 4691-4709, 2024

[11] L. Turchet and P. Casari, "On the Impact of 5G Slicing on an Internet of Musical Things System," in *IEEE Internet of Things Journal*

[12] 3GPP TS 26.253, "Codec for Immersive Voice and Audio Services (IVAS); Detailed algorithmic description incl. RTP payload format and SDP parameter definitions; 3GPP Technical Specification," 2024.

[13] D. G. Malham and A. Myatt, "3-D sound spatialization using ambisonic techniques", *Computer Music Journal*, vol. 19, pp. 58, 1995.

[14] J. Paulus, L. Laaksonen, T. Pihlajakuja, M.-V. Laitinen, J. Vilkamo and A. Vasilache, "Metadata-Assisted Spatial Audio (MASA) - An Overview," *2024 5th International Symposium on the Internet of Sounds,* Erlangen, Germany 2024, in press.

[15] 3GPP TS 26.258, "Codec for Immersive Voice and Audio Services (IVAS); C code (floating-point); 3GPP Technical Specification," 2024.

[16] T. Robotham, A. Singla, O. S. Rummukainen, A. Raake and E. A. P. Habets, "Audiovisual Database with 360° Video and Higher-Order Ambisonics Audio for Perception, Cognition, Behavior, and QoE Evaluation Research," *2022 14th International Conference on Quality of Multimedia Experience (QoMEX)*, Lippstadt, Germany, 2022, pp. 1-6

[17] Capacitor by Ionic - Cross-platform apps with web technology. [Online]. Available: https://capacitorjs.com/

[18] Gradle Build Tool. [Online]. Available: https://gradle.org/

[19] React. [Online]. Available: https://react.dev/

[20] Three.js – JavaScript 3D Library. [Online]. Available: https://threejs.org/

[21] W3C Recommendation, "Web Audio API", June 17, 2021, https://www.w3.org/TR/webaudio/

[22] A. McArthur, C. v. Tonder, L. Gaston-Bird and A. Knight-Hill, "A survey of 3D audio through the browser: practitioner perspectives," *2021 Immersive and 3D Audio: from Architecture to Automotive (I3DA)*, Bologna, Italy, 2021

[23] M. Tomasetti, A. Boem and L. Turchet, "How to Spatial Audio with the WebXR API: a comparison of the tools and techniques for creating immersive sonic experiences on the browser," *2023 Immersive and 3D Audio: from Architecture to Automotive (I3DA)*, Bologna, Italy, 2023

[24] C. van Tonder and M. Lopez, "Acoustic Atlas – Auralisation in the Browser," *2021 Immersive and 3D Audio: from Architecture to Automotive (I3DA)*, Bologna, Italy, 2021

[25] A. Boem and L. Turchet, "Musical Metaverse Playgrounds: exploring the design of shared virtual sonic experiences on web browsers," *2023 4th International Symposium on the Internet of Sounds,* Pisa, Italy, 2023

[26] WebAssembly. [Online]. Available: https://webassembly.org/

[27] Main — Emscripten 3.1.65-git (dev) documentation. [Online]. Available: https://emscripten.org/