IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 3 April 2025, accepted 22 May 2025, date of publication 3 June 2025, date of current version 16 June 2025.

Digital Object Identifier 10.1109/ACCESS.2025.3576167

RESEARCH ARTICLE

Assessing Mulsemedia Authoring Application Based on Events With STEVE 2.0

DOUGLAS MATTOS[1], **RÔMULO VIEIRA**[1], **(Member, IEEE)**,
DÉBORA C. MUCHALUAT-SAADE[1], **(Member, IEEE)**,
AND GEORGE GHINEA[2], **(Member, IEEE)**
[1]MídiaCom Laboratory, Fluminense Federal University (UFF), Niterói 24220-900, Brazil
[2]Department of Computer Science, Brunel University of London, UB8 3PH Uxbridge, U.K.

Corresponding author: Rômulo Vieira (romulo_vieira@midiacom.uff.br)

This work was partially supported by Brazilian Coordination for the Improvement of Higher Education Personnel (CAPES), National Council for Scientific and Technological Development (CNPq), Carlos Chagas Filho Foundation for Research Support of the State of Rio de Janeiro (FAPERJ), and Funding Agency for Studies and Projects (FINEP).

This work involved human subjects or animals in its research. Approval of all ethical and experimental procedures and protocols was granted by the Research Ethics Committee of Fluminense Federal University.

ABSTRACT The concept of Multiple Sensorial Media (mulsemedia) has been explored to enhance user experiences by engaging different senses beyond sight and hearing. The growing demand for developing such applications has led to various studies focusing on the mulsemedia authoring process. However, there remains a significant gap in accessible methods for developing these applications. This article addresses this issue by assessing an event-based approach to mulsemedia creation. We implemented this approach in the graphical authoring tool STEVE 2.0, which enables users to develop mulsemedia applications by defining event-based temporal relationships that synchronize traditional media with sensory effects. Additionally, STEVE 2.0 allows for the configuration of media presentation and sensory effect rendering properties. To assess the usability, features, and user experience of the tool, we conducted experiments with 44 participants. The results show that STEVE 2.0 effectively empowers users to create mulsemedia applications. Furthermore, we discuss the integration of STEVE 2.0 with other multimedia technologies used for interactive digital TV, underscoring the importance of comprehensive platforms that support the entire life cycle of these applications, from production to distribution and rendering.

INDEX TERMS Event-based synchronization, mulsemedia authoring, mulsemedia tools, multisensory applications, sensory effects, STEVE.

I. INTRODUCTION

Multimedia applications are readily available on various devices such as smartphones, computers, tablets, and TV sets. Although these apps predominantly engage the senses of sight and hearing, it is important to recognize that human communication and perception involve all five senses, including touch, taste, and smell. In addition, we have interceptive capabilities that allow us to perceive internal bodily changes, sense of balance, sense of heat and cold, awareness of the position of our body and sense of pain [1].

The concept of Multiple Sensorial Media (mulsemedia) has been introduced to explore and incorporate these various sensory stimuli beyond just sight and hearing. By integrating multiple sensory effects, interactive multimedia applications offer users novel sensations, resulting in a higher quality of experience (QoE) and greater immersion [2], [3], [4], [5], [6]. The use of mulsemedia promises to enhance user engagement and satisfaction with multimedia content, contributing to a more immersive digital experience overall.

Concerning the workflow of mulsemedia applications, we can subdivide it into three phases [7]: authoring (or production), distribution, and rendering of applications in the physical environment. In the authoring phase, which

The associate editor coordinating the review of this manuscript and approving it for publication was Olarik Surinta.

represents the focus of this article, sensory effects can be defined through digital capturing and processing of data obtained from sensors, automatic extraction of sensory effects from audiovisual content, and manual specification by authors. Furthermore, we can combine the manual specification of the authors with a crowd-sourcing approach. In [8], users give suggestions about the time intervals in which specific sensory effects could be activated according to audiovisual contents provided by mulsemedia authors. That approach supports authors in enhancing the specification of time intervals in which sensory effects should be rendered with audiovisual contents.

Regarding advances in mulsemedia application modeling in the authoring phase, studies in [9] and [10] provide sensory effects at a low-level abstraction by identifying sensors and actuators. This approach is frequently used in Internet of Things (IoT) solutions supporting the inherent heterogeneity of devices in this type of environment. However, this approach is not suitable for mulsemedia systems since it does not allow the specification of the spatial and temporal behavior of multimedia content and sensory effects at a high-level abstraction as in traditional multimedia model approaches [11], [12], [13].

Another solution for representing sensory effect metadata is the MPEG-V standard [14], which uses the timeline-based paradigm to synchronize sensory effects with existing multimedia content. Furthermore, other proposals focus on enhancing the specification of sensory effect metadata and the annotation of multimedia content with sensory effects by using graphical tools such as [15], [16], [17]. However, those initiatives also use the timeline-based paradigm, which has several inherent limitations [18]. Moreover, those solutions do not allow authors to specify an entire mulsemedia application by defining media item and effect spatial and temporal behavior. Another approach is that described in [19], which makes use of templates and wizards. However, that solution makes the tool less expressive by restricting authors to the set of predefined applications available in the tool. In addition, the authors are not able to modify the behavior of their applications.

Investigating the challenge of modeling immersive environments with multiple sensory effects is essential for advancing mulsemedia applications. In [7], authors highlight the demand for tools that enhance mulsemedia application development. They also emphasize the importance of the temporal synchronization of sensory effects for supporting the effectiveness of mulsemedia applications.

This context reveals opportunities to improve the development of interactive multimedia applications that include sensory effects. One promising approach is the use of the Multimedia Sensory Effect Model (MultiSEM) to support the creation of graphical environments based on temporal visualizations. This model treats sensory effects as first-class entities, modeling them as nodes within a multimedia application. As a result, sensory effects can be synchronized temporally with other nodes, whether these involve traditional media or other sensory effects. MultiSEM uses an event-based synchronization model, triggering actions based on specific events. These events may be synchronous or asynchronous, depending on their timing. These events take place during the execution of the hypermedia document and can be used to establish conditional temporal relationships between media elements within the document. Furthermore, MultiSEM leverages the concept of a hypermedia connector [20] to represent both spatial and temporal relationships. The model also defines a set of causal connectors to represent temporal relations, drawing on Allen's time interval relations [21]. This high-level abstraction for representing sensory effects was initially proposed in [22].

Given MultiSEM's advantages for multimedia authoring, its principles were integrated into STEVE 1.0 [23], an authoring environment that enables creators to define temporal relationships between different media. This integration led to the development of STEVE 2.0 [24]. Although STEVE 2.0 was introduced in previous studies [24], [25], this updated version of STEVE 2.0 introduces new features, including media presentation configuration and sensory effect rendering. In addition, this updated version includes integration with the Ginga-NCL middleware [26], a Nested Context Language (NCL) presentation engine designed to ensure interoperability between IPTV multimedia application frameworks and serve as middleware for the Brazilian digital TV system. STEVE 2.0 exports applications as NCL 4.0 documents, which support sensory effects [27]. This enables seamless integration with Ginga-NCL and supports the full application lifecycle, from production to rendering.

Based on these new features of STEVE 2.0, this work aims to provide a complete evaluation of the tool's capabilities and usability through a user experiment with 44 participants. The evaluation focuses on key aspects of the user experience, such as the clarity and effectiveness of relating sensory effects to multimedia content, the ease of establishing temporal relationships between different media, and an analysis of interactivity. These dimensions were evaluated using the Goal Question Metric (GQM), System Usability Scale (SUS), and User Experience Questionnaire (UEQ) methodologies. Based on these findings, this study also contributes to the advancement of authoring environments by promoting the creation of tools based on temporal visualizations.

In summary, the main contributions of this work are:

- Extensive user experimentation involving 44 participants to evaluate STEVE 2.0 usability and effectiveness;
- Detailed specification of STEVE 2.0, based on the MultiSEM model, discussing how it is used to create mulsemedia applications by authors with no programming skills;
- Integration of NCL 4.0 applications created with STEVE 2.0 to the Ginga-NCL middleware, incorporating sensory effects into the Brazilian digital TV system and promoting a more immersive viewer experience;
- Addressing the literature research gap in the field of mulsemedia applications by discussing the

limitations of existing approaches and comparing them to STEVE 2.0.

We organize this paper in the following sections. Section II presents related work. Section III describes an interactive mulsemedia application as a case study to show how MultiSEM represents it. Section IV presents STEVE 2.0. We also show how authors can use STEVE 2.0 to create the example application graphically. STEVE's architecture, data flow, and features are also depicted in this section. Section V presents user experiments in order to evaluate STEVE's usability, specific features, and user experience. In addition, it describes the methodologies we have used, the questionnaires for collecting users' feedback, and the analyses of the results. Lastly, conclusions and future work are given in Section VI.

II. RELATED WORK

Our previous work [28] presents a survey on mulsemedia models, languages and authoring tools. In this section, we focus on authoring languages and tools and compare them with STEVE 2.0.

Notably, the MPEG-V standard [14] stands out as a noteworthy technology, providing XML-based elements for specifying real-world objects such as sensors, actuators, and virtual-world entities. These elements enable standardized data interchange between applications of these types and incorporate the Control Information Description Language (CIDL) for portraying metadata related to actuator and sensor capabilities, sensory effects, and user adaptation preferences. The standard utilizes a timeline-based approach to achieve temporal synchronization between sensory effects and audiovisual content, offering fade-in and fade-out options to enhance the QoE of services [22].

Another significant development is NCL 4.0 [29], an extension of NCL 3.0 [26] that integrates sensory effects as first-class entities. This version allows authors to define properties for effect elements and use descriptors to reference them, while also specifying the position of effects in spherical coordinates, making it independent of the application's physical installation. NCL 4.0 additionally facilitates multimodal interaction and supports multiple users [29].

Algebraic System of Aggregates and Mulsemedia data Processing Language (ASAMPL) [30], [31], [32] presents a language specifically designed for representing complex information composed of various data types, including audiovisual, olfactory, gustatory, and environmental data such as temperature, pressure, and density.

In the context of crowdsourcing mulsemedia applications, CrowdMuse [33] serves as a tool to collect sensory effects from the crowd and relay them to service authors. However, its emphasis is primarily on video editing rather than providing a comprehensive authoring environment. H-Studio [34] focuses on supporting two specific sensory effects, namely vibration and movement, providing a timeline for synchronization and a menu for defining effect parameters.

Several tools employ the timeline paradigm for synchronizing sensory effects with audiovisual content. Sensory Effect Video Annotation (SEVino) [15] offers channels representing different types of sensory effects that can be synchronized with video content and supports exportation to MPEG-V format. Similarly, Representation of Sensory Effects (RoSE) Studio [16] provides a graphical interface for defining the spatial position of sensory effects in 3D space and enables exportation to MPEG-V SEM files. Sensible Media Authoring Factory (SMURF) [17] features a timeline for synchronization and supports the creation of groups of effects to compose more complex sensory effects. Real4DAStudio [35] offers a 3D simulation feature for effect validation and employs media-based and event-based interfaces for effect synchronization. MulSeMaker [19] utilizes a model-based interface for non-programmers to define sensory effects and interactivity events, while FeelReal[1] is a commercial tool specializing in scent effects synchronization with video.

The proposal by Jost and Pévédic [36] focuses on the incorporation of multisensory events into audiovisual content through a unified timeline. To achieve this, the authors suggest a shift in authorial perspective and in the way media integration takes place, drawing inspiration from the behavior of musical scores. In pursuit of this objective, such media are divided into various fragments. Each of these segments functions like a musical note, capable of being represented with the same graphical size but with different temporal duration. Consequently, sensory effects are triggered and operate in specific segments, no longer relying on temporal factors or human actions. The primary advantage arising from this functionality is the facilitation of multimedia content programming, making it more accessible as well.

In light of next-generation broadcasting services being focused on delivering realistic media content to enhance user experience through the representation of the five human senses, the work of Jala and Murroni [37] aims to present an updated view of research conducted in the domain of QoE for these types of services. The results are summarized based on subjective quality assessments for audiovisual sequences enriched with effects such as ambient light, wind, and vibration effects. Subsequently, three insights are provided to the broadcasting community, aiding in the understanding of the role of multimedia in applications for this field, namely: i) viewing experience and sense of reality, describing how sensory effects can be applied to the viewing experience and contribute to content enhancement; ii) video quality, demonstrating the significant role sensory effects play in the perceived quality of the video; and iii) sensory effects and emotions, emphasizing that the use of effects can intensify emotions.

The proposals by Coelho et al. [38], [39] aim to introduce an authoring tool featuring three distinct interfaces (desktop, immersive interface, and tangible interface) for the creation

[1] https://qvarta.com/en/works/feelreal-sensory-mask

of 360° multisensory videos. One notable advantage is the provision of a live preview of the multisensory content being produced [38]. Additionally, they propose a new architecture for an immersive collaborative tool enabling the real-time creation of multisensory Virtual Reality (VR) experiences through two authoring tools (Desktop Interface and Immersive Interface). This promotes swift development, collaborative creation, and the construction of customized solutions, optimizing and enhancing the efficiency of the editing process [39].

AMUSEVR [40] is an immersive authoring and presentation environment for creating 360° interactive mulsemedia applications. AMUSEVR exports applications in MultiSEL, an XML-based language that is also based on the MultiSEM conceptual model. Different from STEVE 2.0, which provides a traditional 2D user interface, AMUSEVR runs on Oculus Quest and focuses on 360° content.

Table 1 summarizes a comparison among mulsemedia authoring tools found in the literature and STEVE 2.0, which is our proposal. We can notice that STEVE 2.0 uses the event-base synchronization paradigm, provides a graphical user interface for temporal and spatial views, provides synchronization of multiple media and sensory effects, provides the authoring of interactivity relations, was designed for non-expert users, provides sensory effect automatic extraction [25], [41], exports applications into NCL 4.0 format, provides error analysis during authoring and also a preview frame to check the application execution. We can notice from the table that no tool provides support for statement assessment relations to evaluate state variables.

III. USING MULTISEM TO SPECIFY AN INTERACTIVE MULSEMEDIA APPLICATION

This section describes an example of mulsemedia application, which is based on the application presented in [22]. The application describes an immersive environment with multiple sensory effects for cognitive activities with children with autism. It aims to demonstrate how MultiSEM represents the application. In this interactive application, users are sat on an armchair in the center of a room. There are actuators and sensors spread in the immersive environment. There is also a screen to display the audiovisual content that is synchronized with sensory effects.

Figure 1 illustrates the initial scenario, where the screen displays a living room with a burning fireplace (*background_image*). After 5 seconds, a video and audio depicting the fireplace start playing, along with a light effect to represent the flames' luminosity. Another 5-second delay triggers a temperature effect, simulating the heat from the fireplace.

In MultiSEM, the scenario is represented by modeling the background image as a media node (*Media* class) and the light and temperature effects as *SensoryEffect* nodes. The light effect's *effectPresentationProperty.intensity* is set to 50 lux, and the temperature effect's intensity attribute is set to 32 Celsius degrees. Temporal synchronization is achieved with a synchronization relation that uses a *CausalConnector* with a simple condition role for the *presentation* start event and an *ActionRole* for the *start* action. A delay attribute of 5s is defined to fire the action.

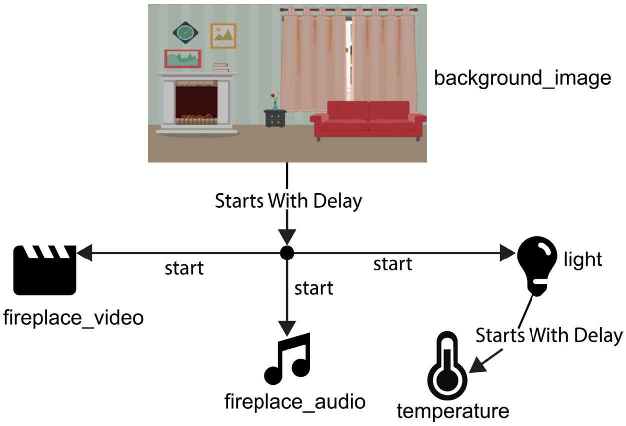


FIGURE 1. Structural view of the multimedia application developed from MultiSEM [22].

In the second scenario (Figure 2), user interaction, such as pressing the green key on a remote control, triggers the start of two media items (window opening and outside view) and three sensory effects (light, wind, and temperature). A statement assessment evaluates whether the window is open or closed (*windowsOpen=1* or *windowsOpen=0*). If the window is closed, the interactivity relation sets *windowsOpen=1* and activates the media items and sensory effects.

FIGURE 2. Mulsemedia application after user interaction [22].

The connector's glue uses a compound trigger expression with an *and* operator to activate the relation actions (*start* and *set*) based on user interaction and statement assessment. The condition role is linked to the *background_image* item through the selection event (*green_key*). Another condition represents the statement assessment, connected to the *windowsOpen* node. The relation specifies *start* and *set* actions, associated with presentation and attribution events, respectively. The *startAction* triggers *window_video* and *window_audio* items, along with sensory effects. The *setAction*

TABLE 1. Comparison of mulsemedia authoring tools.

		STEVE 2.0	CrowdMuse [8]	H-Studio [34]	Kim et al. [42]	SEVino [15]	RoSE [16]	SMURF [17]	Real 4D Studio [35]	MulSeMaker [19]	FeelReal
Temporal Sync. Paradigm		E	T	T	T	T	T	T	T	E & T	T
Authoring GUI Approach		T & S	W	T	T	T	T	T	T	W	T
Sync. of Multiple Media and Sensory Effects		✓								✓	
Spatial View Editing	SE Position Editing	✓		+/-		✓	✓	✓	✓		+/-
	Rendering Editing	✓		✓	+/-	✓	✓	✓	✓	+/-	+/-
	Multimedia Spatial View Editing	✓									
Asynchronous Events	Interactivity Relations	✓								+/-	
	Statement Assessment Relations										
Ordinary User Support		✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
SE Automatic Extraction		✓			+/-				✓		
Publishing Formats		NCL 4.0	MV	none	MV	MV	MV	MV	MV	HTML5	none
Error Analysis		✓									
Preview or Simulator or Player		✓				✓			✓		✓

+/- (partially supported feature); ✓ (fully supported feature);
SE (Sensory Effect)
Temporal Synchronization Paradigm: E (event-based); T (timeline-based);
Authoring GUI Approach: T (temporal); S (spatial); W (wizard)
Publishing Formats: MV (MPEG-V)

role includes the *pValue* parameter for *windowsOpen*. Both actions are triggered simultaneously using the *par* operator in the glue expression.

Figure 3 illustrates the second scenario of our mulsemedia application example to show how MultiSEM represents interactivity events and statement assessments.

IV. STEVE 2.0-AN AUTHORING ENVIRONMENT FOR MULSEMEDIA APPLICATIONS

In this section, we present STEVE 2.0 [24], our proposed multimedia authoring tool, which functions as a proof of concept for the implementation of the MultiSEM model within mulsemedia applications. Additionally, we examine the architecture, data flow, and graphical user interface of STEVE 2.0. Finally, we emphasize the tool's integration with other mulsemedia technologies, which collectively offer a comprehensive end-to-end toolchain for authoring and rendering mulsemedia applications.

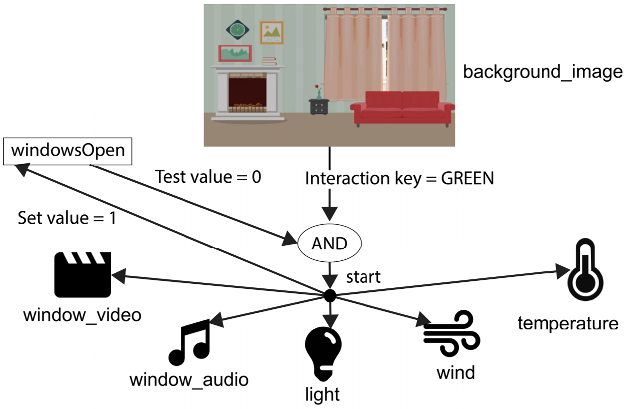


FIGURE 3. Structural View of Scenario 2 [22].

A previous version of STEVE [43] was available for authoring multimedia applications. This version was also designed to simplify content authoring for users without knowledge of multimedia authoring languages and models, but it was initially intended for the development of hypermedia applications for digital TV and web systems. Its temporal synchronization model was based on events within the Simple Interactive Multimedia Model (SIMM), which was capable of providing temporal, spatial, and interactive relationships. This resulted in a tool for editing spatio-temporal visualizations of hypermedia documents, allowing authors to create causal temporal relationships between media items. The editor also supports the definition of viewer interactions and the simulation of these asynchronous events to preview the document presentation. Besides that, users can define media presentation properties and verify them within the spatial visualization interface. The current implementation, named STEVE 2.0, extends the original version to support the integration and synchronization of sensory effects with traditional multimedia content. Furthermore, this new model includes task automation, as introduced in [40] and evaluated in [41], in addition to introducing a new paradigm for event synchronization, the MultiSEM.

Since STEVE 2.0 is based on MultiSEM, it uses an event-based temporal synchronization paradigm to define the temporal behavior of mulsemedia applications. Its authoring GUI approach is based on temporal view, as discussed in IV-D, to allow users with no programming skills to define mulsemedia applications. The tool graphically provides causal temporal relations based on MultiSEM's relations and gives authors feedback about temporal synchronization inconsistencies. In addition, users can create interactivity relations to activate, for example, sensory effects due to user interactions with the mulsemedia application. The editor also provides spatial view editing to allow users to edit rendering characteristics (e.g., intensity, scent type, light frequency) and physical positions of sensory effects. Moreover, we can edit the presentation properties of traditional multimedia content.

Furthermore, STEVE 2.0 allows authors to check the temporal and spatial behavior of mulsemedia applications by providing a graphical temporal and spatial view in a synchronized way. After the production phase, STEVE mulsemedia applications can be distributed by exporting them to documents written in NCL 4.0 [22], which can run in the multisensory-extended Ginga-NCL middleware presented in [29].

A. ARCHITECTURE

Figure 4 shows the STEVE 2.0 modular architecture[2] highlighting the new components added (green boxes) to STEVE 1.0 [43] and which components were updated (yellow boxes) to support mulsemedia applications. The architecture uses the Model-View-Controller (MVC) design pattern, which separates the system into three modules that communicates with each other: *View*, *Model* and *Controller*. The first one is responsible for implementing the graphical user interface of STEVE, getting model data from *Model*, and notifying *Model* about user interactions with *View* using the *Controller* option. *Model* describes the STEVE business logic part, which implements data retrieval, and MultiSEM's event-based synchronization paradigm and its entities. Also, it notifies *View* about model changes in order to update STEVE's GUI.

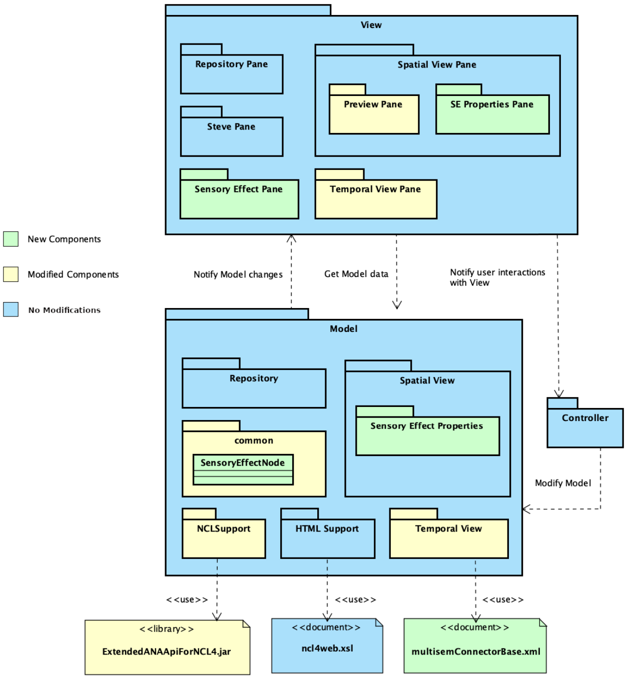


FIGURE 4. STEVE 2.0 architecture highlighting the modifications against STEVE 1.0.

In *Model*, we have the STEVE extension core to support mulsemedia applications, the class *SensoryEffectNode*, which implements sensory effects as first-class entity following MultiSEM representation. In addition, we have added the new package *Sensory Effect Properties* into *Spatial View* to represent the rendering characteristics of sensory effects according to MPEG-V's specification, on which MultiSEM is based. Moreover, the *Temporal View* package was modified to use MultiSEM's temporal synchronization entities. Additionally, STEVE 2.0 uses MultiSEM's predefined connector base (*multisemConnectorBase.xml* artifact) to represent the temporal causal relations it provides. We have also extended ANA API [44] in order to export STEVE 2.0 projects into NCL 4.0 documents. ANA API is a metamodel specifically created to represent NCL documents in model-oriented environments enhancing the NCL code manipulation.

The *Repository* and *HTML Support* packages, also presented in *Model*, were not modified. The first one implements the data and business logic for STEVE's repository, in which users manipulate multimedia files. *HTML Support* is responsible for exporting STEVE projects with no sensory effect content, i.e. only interactive multimedia applications, into HTML5 documents. The package uses

[2] https://github.com/dougpmattos/steve

the NCL4Web tool (*ncl4web.xsl* artifact), which uses XSLT and JavaScript libraries, to transform NCL documents into HTML5 applications.

Regarding the *View* module, STEVE 2.0 has added the *Sensory Effect Pane* and *SE Properties Pane* modules. The first one implements a graphical control element that provides the list of sensory effects defined in MultiSEM. The *SE Properties Pane* implements a pane to allow users to edit sensory effect rendering properties according to the effect type. In addition, we have modified the *Preview Pane* package to help authors to visualize the temporal synchronization between sensory effects and audiovisual content. Furthermore, *Temporal View Pane* was updated to create the graphical representation of sensory effects in STEVE's temporal view. We will discuss those graphical elements for sensory effects in Section IV-C.

B. DATA FLOWS

STEVE's data flow diagram, presented in Figure 5, shows three input types the editor can receive: new projects, pre-existing STEVE 2.0 projects and NCL 3.0 [26] documents. The STEVE project serialized file is interpreted in order to recover MultiSEM entities. When NCL 3.0 documents are imported into STEVE, they are mapped into MultiSEM entities through processes 1, 2, and 3 indicated in Figure 5.

Process 1 transforms the NCL document into a graph model called Hypermedia Temporal Graph (HTG) [45]. HTG represents the multimedia application temporal behavior in a digraph structure. It contains all predictable and unpredictable events, which can modify NCL media item presentation event states (*sleeping*, *paused* and *occurring*). In *Process* 2, STEVE generates another data structure called *Presentation Plan* [45] to indicate event execution times. Finally, in *Process 3*, the NCL document is represented using MultiSEM entities. Notice that the original NCL links from the imported document are not directly mapped into MultiSEM relations, except the user interaction relations. Instead, STEVE creates *Starts with Delay* relations to temporally synchronize the document nodes according to their execution times defined in the presentation plan.

In order to export STEVE applications to NCL documents, the tool transforms MultiSEM entities into NCL entities, shown in *Process 5* in Figure 5, using the ANA API [44], which represents NCL elements as Java classes. The editor also uses ANA methods to write NCL documents in *Process 6*. We have extended ANA API to support the sensory effect node present in NCL 4.0. Therefore, STEVE 2.0 exports its applications to NCL 3.0, ready to run in Digital TV and IPTV systems [26], and NCL 4.0 [29], being able to run in an extended Ginga-NCL for sensory effects [27], which is adopted for the Application Coding Layer of the next generation digital TV System in Brazil (TV 3.0).[3] STEVE also allows exporting applications to HTML5 documents. In this case, first, the editor generates the NCL 3.0 document and stores it (*Data Store D1*). Then STEVE uses NCL4Web [46] to write the HTML5 document (*Process 7*) representing the stored NCL document. Moreover, STEVE applications that are opened in the editor can be saved (*Process 4*) in STEVE file format to be available for future editing.

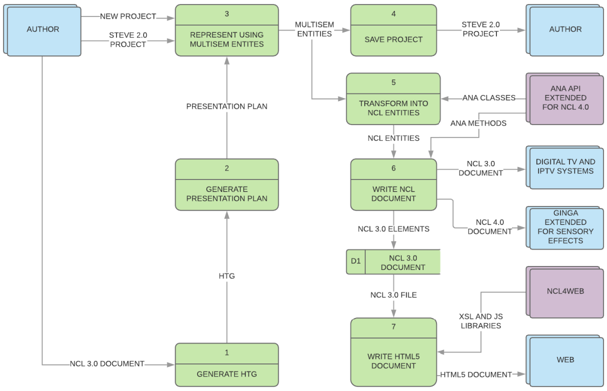
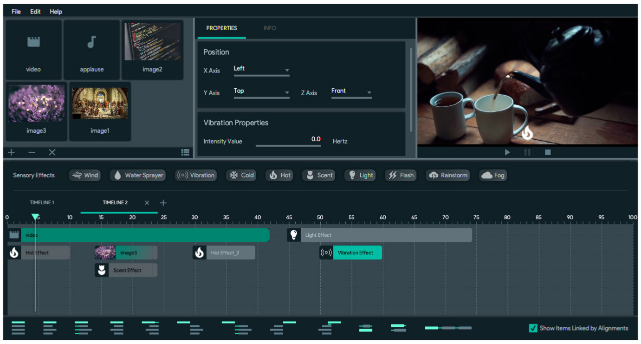


FIGURE 5. STEVE 2.0 Data flow.

C. GRAPHICAL USER INTERFACE

Figure 6 shows the interface of STEVE 2.0 to support multiple sensory effects. The interface of STEVE consists of a multimedia content repository in the upper left corner, a panel in the middle for users to edit media presentation properties and sensory effect rendering characteristics, a preview pane to play audiovisual content in the upper right corner, and a temporal view in the bottom region. The pane for editing the rendering of sensory effects is based on the MultiSEM effect properties specification. In addition, STEVE 2.0 provides a graphical control element with a list of sensory effects above the temporal view, as shown in Figure 6.



Position

X Axis Left

Y Axis Top Z Axis Front

Vibration Properties

Intensity Value 0.0 Hertz

☑ Show Items Linked by Alignments

FIGURE 6. Graphical interface of STEVE 2.0.

Users can select one of those sensory effects and drag them to the timeline to temporally synchronize them with other nodes. To do that temporal synchronization, users can use the causal temporal relations STEVE graphically provides below the timeline, as shown in Figure 6. Notice that some sensory effects available in the STEVE interface are not defined in

[3] https://forumsbtvd.org.br/tv3_0/

MultiSEM since they are implemented by grouping different effects defined in MultiSEM. For example, the *Rainstorm* effect is mapped into the *Flashlight*, *Fog*, *Wind* and *Rain* effects.

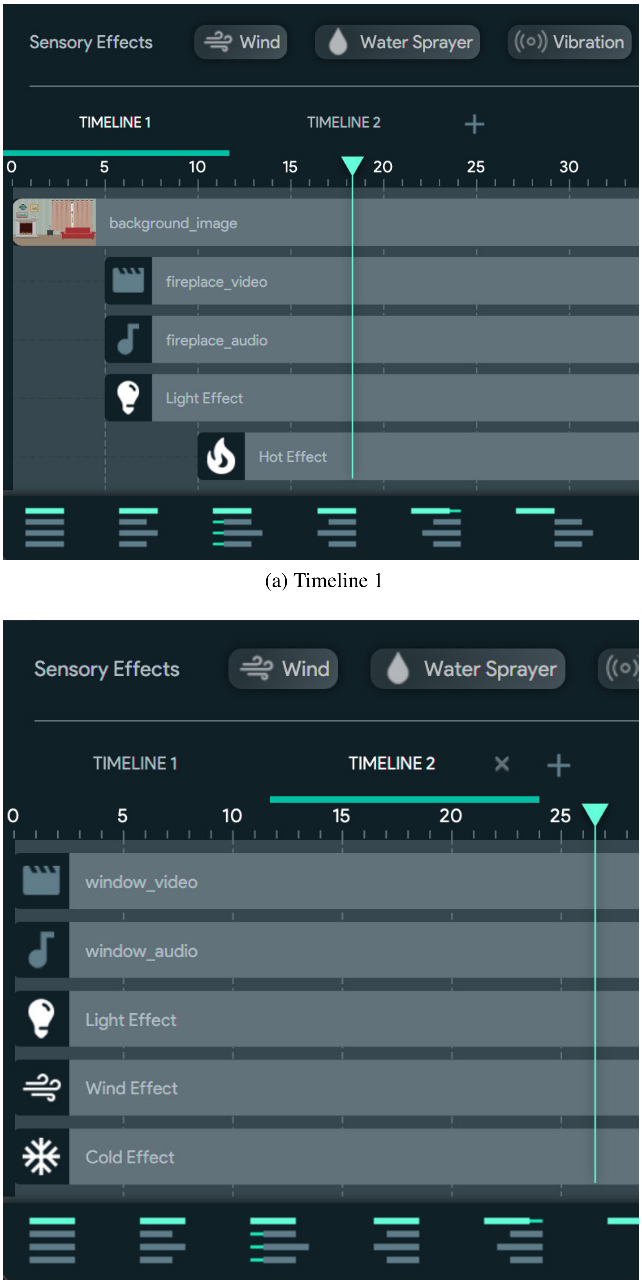
Moreover, the preview pane also displays icons that represent sensory effects temporally synchronized with audiovisual content. For instance, in Figure 6, a heat effect is shown in the preview frame synchronized with the first video of the application, where a coffee is served.

D. TEMPORAL VIEW

To support authors to synchronize their mulsemedia applications, STEVE provides causal temporal relations graphically in the button bar below the timeline, as shown in Figure 6. STEVE 2.0 uses the part 1 of MultiSEM's predefined connector base (*Allen's Temporal Relations*) to define those relations.

Figure 7a shows the temporal view in STEVE 2.0 that corresponds to the structural view of the example discussed in Section III and shown in Figure 1. To create this temporal view in STEVE 2.0, authors should first drag the media and sensory effect nodes from the media repository and the sensory effect bar, respectively, to the timeline. After that, they should synchronize them using the temporal relation buttons.

In addition to supporting synchronous temporal relations, STEVE 2.0 allows for the definition of asynchronous relations to facilitate user interactions. To create the interaction illustrated in Figure 3, authors must specify the interaction key (e.g., a keyboard key, remote control button, or any interaction device that triggers an action within the mulsemedia application) and determine which media items or sensory effects within the application will stop or start following the user's movements. In our example, the authors need to designate the *Green* key as the interaction trigger and select the *window_video*, *window_audio*, *light*, *wind*, and *hot* (temperature) nodes to initiate upon the user's press of the *Green* key. After defining that interactivity relation, STEVE automatically creates a new timeline, *Timeline 2* as shown in Figure 7b, to contain the new nodes that start. Therefore, for each interactivity relation defined, STEVE 2.0[4] creates a new event-based timeline.



(a) Timeline 1

(b) Timeline 2

FIGURE 7. STEVE 2.0 Event-based timelines for our mulsemedia application.

E. STEVE 2.0 INTEGRATED WITH OTHER MULSEMEDIA PLATFORMS

This section discusses how STEVE 2.0 is integrated with another mulsemedia platform to provide an end-to-end tool chain as a complete mulsemedia platform accomplishing the three phases of the development of mulsemedia applications [7]: authoring, distribution, and rendering in the physical environment.

STEVE 2.0 was integrated with a machine learning solution to enhance visual content's sensory effect annotation. That content-driven component, named STEVEML[5] (STEVE Machine Learning) [25], gives STEVE 2.0 the ability to automatically extract sensory effects from video content. With that integration, authors can select video or image items in STEVE's temporal view and request the sensory effect annotation for the selected media according to the effect types chosen. After the extraction, authors can also make manual adjustments in the temporal synchronization of the extracted effects.

[4] Demo Videos for STEVE 2.0 Main Features: https://dougpmattos.github.io/#13

[5] Demo Video for STEVEML: https://dougpmattos.github.io/#28

Concerning the distribution and rendering phases, STEVE 2.0 was integrated[6] with the multisensory Ginga-NCL proposed in [27] by exporting STEVE projects to documents written in NCL 4.0 [27], ready for the next generation digital TV system in Brazil. That Ginga-NCL 4.0 mulsemedia formatter implements an effect renderer for each effect type and implements three APIs *DeviceScent*, *DeviceWind* and *DeviceLight* to render scent, wind, and light effects respectively.

F. STEVE 2.0 USE CASE EXAMPLES

STEVE 2.0 enables the authoring of multisensory applications by synchronizing traditional media with sensory effects. Its event-based approach offers flexibility and ease of use, making it applicable to several domains, including education, entertainment, and healthcare. Below, we outline four hypothetical cases that illustrate its potential. These examples do not claim to represent actual user requirements but serve as a starting point for discussions about how STEVE 2.0 could be applied and which features could be further developed to enhance its impact.

1) EDUCATION

STEVE 2.0 can be used to create immersive educational applications that enhance student engagement and retention by incorporating multisensory stimuli. Examples include:

- **Virtual Science Experiments:** A chemistry lesson where students watch a reaction on screen while feeling a subtle heat effect to simulate exothermic reactions;
- **History and Cultural Education:** A virtual tour of historical sites where users experience ambient sounds, vibrations (simulating footsteps on cobblestone streets), or even scent effects to evoke ancient settings;
- **Foreign Language Learning:** Associating words with sensory experiences, such as showing a beach scene while playing wave sounds and diffusing a light sea breeze aroma effect.

2) ENTERTAINMENT

STEVE 2.0 can be leveraged in interactive storytelling and multimedia production to create more immersive entertainment experiences. Possible applications include:

- **4D Movies:** Enhancing films by synchronizing real-world effects with key cinematic moments (e.g., wind effects during storm scenes, heat effects during desert sequences);
- **Interactive Narratives:** Involving the viewer to decide how a movie story goes on, like the famous Netflix Black Mirror: Bandersnatch, where the viewer can interact to chose the next part of a story;
- **Augmented Concert Experiences:** Synchronizing concert footage with environmental effects, such as vibrations matching bass frequencies or light effects synchronized with stage lighting.

3) HEALTHCARE

Multisensory environments have been used in healthcare and therapy, particularly for patients with cognitive impairments, sensory processing disorders, and mental health conditions [47]. STEVE 2.0 can support:

- **Cognitive and Sensory Therapy:** Creating personalized therapy sessions for individuals with autism or sensory processing disorders by controlling specific sensory inputs in response to user interactions;
- **Pain and Stress Management:** Designing relaxation applications that synchronize calming visuals with wind effects, gentle light shifts, and scents to reduce anxiety and stress;
- **Rehabilitation Training:** Helping stroke patients recover motor functions by associating movement exercises with multisensory feedback, reinforcing progress through haptic and visual cues.

4) DIGITAL TV AND INTERACTIVE BROADCASTS

Given STEVE 2.0's integration with NCL 4.0 and Ginga-NCL, it can be utilized for interactive TV applications, such as:

- **Mulsemedia-Enhanced TV Shows:** Enriching nature documentaries by integrating effects such as wind during storm sequences or temperature variations in climate-related segments;
- **Interactive Cooking Shows:** Adding scent diffusion and vibration effects to enhance engagement, allowing users to experience food preparation in a more immersive way;
- **Educational TV Programs:** Making learning experiences more dynamic by introducing synchronized sensory effects during educational segments.

V. EVALUATION

This section presents an analysis of the usability of STEVE 2.0 in the context of mulsemedia application authoring. The methodology utilized in the testing phase is also outlined, detailing the questionnaires employed for gathering user feedback. Following this, the collected data is thoroughly analyzed, accompanied by a discussion of the tool's operability, alongside the limitations and challenges encountered during the process.

In the experiments, 44 users participated, of which 35 were Computer Science students, and 9 were from other fields, such as Cinema, Medicine, Mathematics, Physics, History, and Law. Three distinct methodologies were employed to evaluate STEVE: Goal Question Metric (GQM) [48], System Usability Scale (SUS) [49], and User Experience Questionnaire (UEQ) [50]. The following sections present the tasks performed and how each of these techniques was applied to the STEVE experiments.

[6] STEVE application running on the multisensory-extended Ginga-NCL: https://bit.ly/3DOu6hW

TABLE 2. STEVE's experiment goals.

Goals	Description
G1	Analyze STEVE's manual synchronization between sensory effect and traditional media for the purpose of evaluation with respect to perspicuity and effectiveness from the point of view of the users.
G2	Analyze STEVE's temporal relations for the purpose of evaluation with respect to perspicuity and effectiveness from the point of view of the users.
G3	Analyze STEVE's interactivity relation feature for the purpose of evaluation with respect to perspicuity and effectiveness from the point of view of the users.
G4	Analyze STEVE for the purpose of evaluation with respect to its overall usability from the point of view of the users.
G5	Analyze STEVE for the purpose of evaluation with respect to users' experience.

A. METHODOLOGY

The GQM framework can be visualized as a directed graph, where the flow progresses from goal nodes to question nodes, and finally to metric nodes. Furthermore, the GQM guidelines emphasize defining both the purpose and the perspective of the goals. The purpose outlines the object of study and the rationale behind its analysis, while the perspective specifies the particular angle or aspect of evaluation and identifies the source or stakeholder providing that evaluation.

Table 2 outlines the objectives established for evaluating STEVE, all of which focus on its temporal synchronization feature (the temporal view). In the "Description" column, each objective specifies the object of study (e.g., G2 - STEVE's temporal relations), the purpose of the analysis (e.g., G2 - evaluation), the aspect being assessed (e.g., G2 - perspicuity and effectiveness), and the perspective from which the assessment is made (e.g., G2 - users). The set of questions corresponding to each goal is detailed in Appendix. For G4, we employed the System Usability Scale (SUS) questionnaire [49], while for G5, we used the User Experience Questionnaire (UEQ) [50]. For the remaining goals, we designed a specific questionnaire tailored to each objective.

In our evaluation, we use the term *effectiveness* in G1 and G2 according to the definition in [49]. It means the ability of users to complete tasks using the system and the quality of the output. The term *perspicuity* that we also use in those goals follows the definition presented in [50]. It is related to the ease of understanding of the evaluated product.

To ensure the usability testing of STEVE was accessible to users, we developed an online presentation[7] designed to guide participants through each step of the process. This approach enabled users to conduct the experiments remotely, without requiring external assistance. The presentation begins by introducing the concept of multisensory applications and encouraging user participation in the study. Next, the core functionalities of STEVE are presented through brief instructional videos, allowing participants to familiarize themselves with the tool. The main features highlighted are as follows:

- Import media files to STEVE;
- Drag and drop media and sensory effects into the temporal view;
- Define the duration of sensory effects and media items;
- Preview STEVE's mulsemedia applications;
- Create temporal alignments to synchronize items;
- Create interactivity events.

Subsequently, we provided users with instructions to download and install STEVE, which is available for macOS[8] and Windows.[9] Prior to beginning the tasks, the necessary assets, such as image and video files, were also supplied to participants. Upon completing each task, users were asked to upload the STEVE project they had created and to provide feedback by completing the corresponding questionnaires, which are detailed in Appendix. It is important to note that user feedback was considered valuable, regardless of whether they successfully completed the task. After the final task, participants were requested to fill out the SUS and UEQ questionnaires, also available in Appendix. All questionnaires were distributed via Google Forms, and the entire experiment was conducted remotely.

1) TASKS

Task 1 proposes a simple mulsemedia application with two sensory effects, hot and cold, and a video media object. Then, the task goal is to synchronize both sensory effects with the video scenes. Also, users were not allowed to use the automatic extraction of sensory effects [25] and the temporal alignments STEVE provides graphically. That is, users must synchronize the items manually by dragging the effect and media items into the STEVE's temporal view, as presented in Section IV-D.

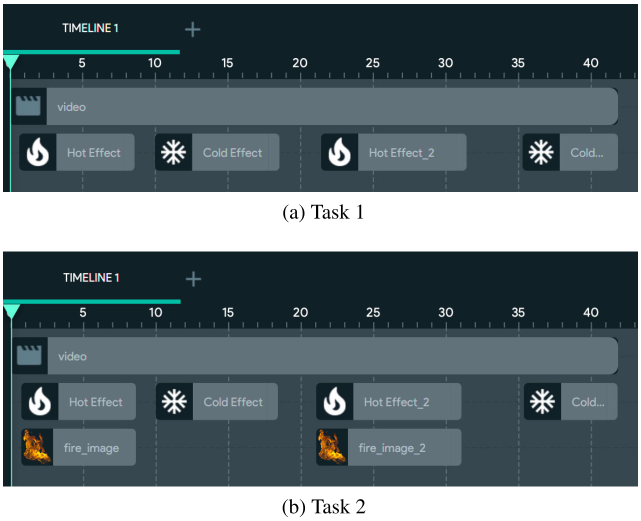
Task 2 adds a new image media into the mulsemedia application introduced in Task 1. In this task, users must show that image, which represents fire, together with the hot sensory effects created in Task 1 using the graphical temporal alignments STEVE provides. Figure 8 shows the temporal view of a mulsemedia application created to perform Task 1 and the result after completing Task 2.

Task 3 proposes another application in which users must create an interactivity relation using three items: an image representing an interactive button and two video files that represent a movie trailer and the movie itself. First, to perform the task, users must show the button image together with the trailer video. In other words, the image needs to be presented when the video starts playing and stops its presentation when the video ends. Then, users must make the button image interactive by defining that when the *ENTER* key is pressed, the application presents the movie video and stops the trailer. Figure 9 shows the temporal view of an application built to perform Task 3. That temporal view contains two timelines, and the second one is derived from the interactivity defined in Timeline 1.

[7] https://dougpmattos.github.io/

[8] https://bit.ly/3BwjvGO

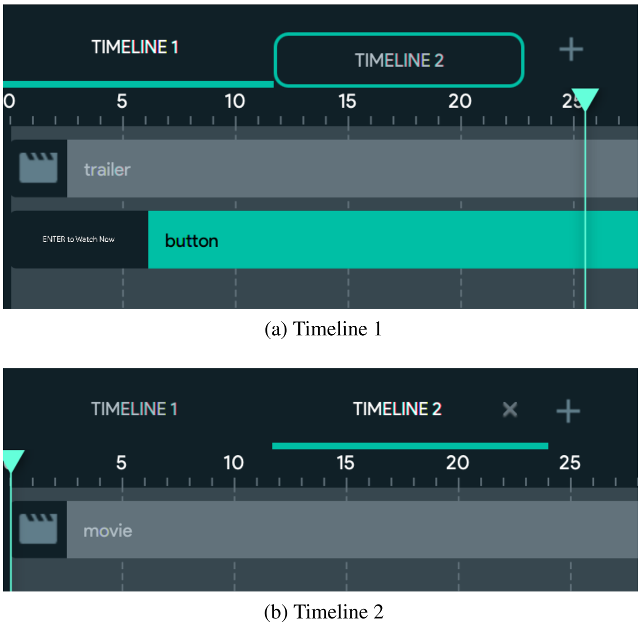
[9] https://bit.ly/3gRICM7



(a) Task 1

(b) Task 2

FIGURE 8. STEVE's temporal view for task 1 and 2.



(a) Timeline 1

(b) Timeline 2

FIGURE 9. STEVE's temporal view for task 3.

B. RESULTS

This section presents an analysis of the data collected from the questionnaire responses, focusing on the evaluation of STEVE's usability, user experience, and the effectiveness and clarity of its features. For the questionnaires corresponding to goals *G1*, *G2*, and *G3*, the mean value of each question was calculated to assess the respective objectives. The SUS and UEQ methodologies were employed to evaluate *G4* and *G5*, respectively. For SUS, the final score was calculated, while for UEQ, we utilized the UEQ data analysis tool,[10] which computes the mean value for each scale of the UEQ.

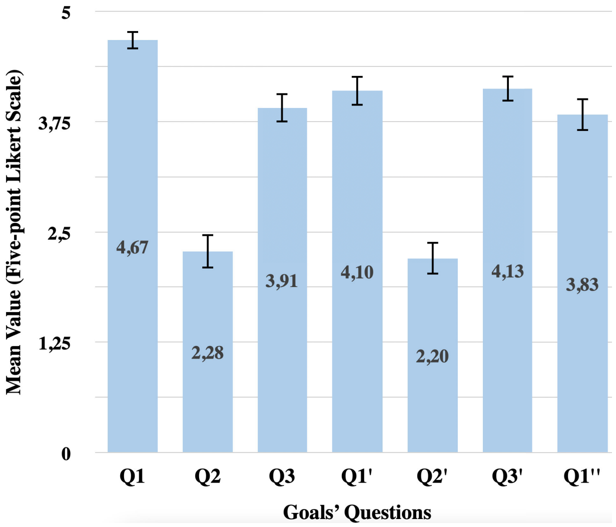


FIGURE 10. Mean value for *G1*, *G2*, and *G3* questions.

1) G1 ANALYSIS

We define two metrics to achieve our *G1* goal (*Analyze the STEVE's manual synchronization between sensory effect and traditional media for the purpose of evaluation with respect to perspicuity and effectiveness from the point of view of the users*): *PP* (Perspicuity) and *EF* (Effectiveness). The first one is measured by calculating the mean value for the *Q1*, *Q2*, and *Q3* G1 question responses using a five-point Likert scale. Therefore, a mean value greater than 2.5 for one of those questions defines that users agree, on average, with its statement.

Figure 10 illustrates the mean values (vertical axis) for the questions related to *G1*, *G2*, and *G3* (horizontal axis). The results reveal that *Q1* received the highest mean score (4.67), suggesting that participants found STEVE's graphical representation of sensory effect types easy to comprehend. In contrast, the mean score for *Q2* (2.28) indicates that, on average, users disagreed with the statement "I found changing sensory effect duration hard in STEVE". Finally, *Q3* shows that, on average, users found synchronizing sensory effects with video scenes relatively straightforward, with a mean score of 3.91.

Regarding the *Q4* question, the *EF* metric is measured using the number of positive answers supported by the quality of the users' STEVE project for Task 1. In the experiment, all 44 users completed Task 1 and had their STEVE projects classified as successful. We can thus evaluate that the STEVE's manual synchronization between sensory effect and traditional media is effective from the users' point of view.

Therefore, we can evaluate that STEVE's manual synchronization between sensory effect and traditional media is perspicuous and effective from the point of view of the users, achieving our goal *G1*.

2) G2 ANALYSIS

Regarding the *G2* goal (*Analyze the STEVE's temporal relations for the purpose of evaluation with respect to*

[10] Link to the UEQ Data Analysis Tool: https://www.ueq-online.org/Material/Short_UEQ_Data_Analysis_Tool.xlsx

perspicuity and effectiveness from the point of view of the users), it also uses the same G1 metrics, *PP* (Perspicuity) and *EF* (Effectiveness). According to Figure 10, itQ1' has obtained a high mean value of 4.10, suggesting that users found the temporal synchronization between media and sensory effects easy to use. For *Q2'*, we have obtained a mean value of 2.20, indicating users did not find the temporal alignments hard to use on average. *Q3'*, with a high mean value of 4.13, points out that users found the temporal alignments icons easy to understand. We can thus evaluate that STEVE's temporal relations are perspicuous from the users' point of view.

Since *Q4* also makes part of *G2*, we questioned users whether they managed to complete *Task 2* successfully. Among 44 users, only one could not complete the task, and those who completed provided succeeded STEVE projects. In addition, with *Q4'*, we collected which temporal alignments users made use. Most of them used the *Equal* temporal relations to synchronize the Task 1 image media with the sensory effects (users had to show an image together with the sensory effects). It also indicates that users understood how to use the temporal alignments. Therefore, we can evaluate that the STEVE's temporal relations are effective from the point of view of the users, achieving our goal *G2*.

3) G3 ANALYSIS

Figure 10 shows the *Q1"* question for *G3* (*Analyze the STEVE's interactivity relation feature for the purpose of evaluation with respect to perspicuity and effectiveness from the point of view of the users*). It obtained a mean value of 3.83, pointing out that users found the interactivity functionality easy to use. However, that mean value was the lowest among the others and nine users (three from non-technological area) could not complete Task 3 successfully, suggesting that STEVE's interactivity feature can be improved. Although the results indicate that the interactivity feature needs improvement, especially for non-technical users, most participants were able to use it successfully, suggesting acceptable clarity and effectiveness overall, achieving our goal *G3*.

4) SUS SCORE FOR G4

To achieve the *G4* goal (*Analyze STEVE for the purpose of evaluation with respect to its overall usability from the point of view of the users*), we have applied the SUS questionnaire. That methodology defines SUS Score for each response to its questionnaire to represent a measure of the overall usability [49]. Therefore, we calculated the SUS Score for each user and the SUS Score average among users.

We have obtained *76.62* as SUS Score average for the STEVE's experiments with a standard deviation of 12.48. That score is above average (68) [51]. Therefore, STEVE's overall usability can be classified as *acceptable* [52] in a range from 0 to 100, in which under 50 is *unacceptable*, between 50-70 is *marginally acceptable*, and above 70 is *acceptable*. Thus, we have achieved the *G4* goal evaluating the STEVE's overall usability.

In addition, we can classify the STEVE's usability as *Good* in an adjective scale [52] that is divided into six categories. Figure 11 shows those scales associated with raw SUS Score, in which the letter "E" means *Excellent* (between 80 and 85 in the SUS Score scale). Thus, we can evaluate that STEVE's temporal relations are effective from the point of view of the users, achieving our goal *G2*.

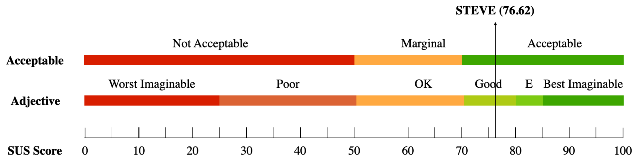


FIGURE 11. Acceptability and adjective scales associated with raw SUS score.

5) UEQ ANALYSIS TOOL FOR G5

For goal *G5* (*Analyze STEVE for the purpose of evaluation with respect to users' experience*), we have analyzed the UEQ questionnaire filled out by users after completing all tasks in the experiment. That methodology has six scales and, for each one, it gives us a mean value calculated from the user responses for the 26 pairs of opposite terms.

UEQ does not produce an overall score for the user experience. Instead, we have a mean value for each scale so that we can interpret it correctly. Figure 12 shows the mean value for each UEQ scale with the respective variances we obtained in the STEVE's experiments. The range of the scales is between −3 (horribly bad) and +3 (extremely good). For all scales, except *Dependability*, STEVE was classified as *Excellent* compared with the benchmark presented in [50]. That benchmark has a data set that contains data from 468 studies regarding different products. *Excellent* means that the obtained result is in the range of the 10% best results among those studies.

The *Perspicuity* and *Efficiency*, which we consider the most important UX aspects for STEVE, obtained a high value of *2.012* and *2.073* respectively. In addition, *Attractiveness* and *Simulation* also obtained high values of *2.13* and *2.00* respectively. We can thus conclude that STEVE is easy to understand and allows users to solve their tasks without unnecessary effort from the point of view of the users' experience.

For the *Dependability* scale, STEVE was classified as *Above Average*, 25% of results better, 50% of results worse. This result suggests that, in general, users perceived STEVE 2.0 as a reliable tool, but that, in certain situations, its behavior may have been less predictable than expected. Some factors that may have contributed to this perception include: i) the event-based operating model, which may have represented a challenge for users less familiar with this paradigm, making it difficult to define interactions and dependencies between multimedia elements and sensory effects; ii) the absence of user interface tooltips, which may have made some operations less intuitive; and iii) the complexity of the interactivity configuration, which may have impacted the perception of control and predictability of the tool.

Some aspects of the tool's functionalities can be improved to improve user confidence and reduce potential uncertainties, such as enhancing visual feedback, allowing a clearer representation of relations between events and synchronization status; automated error detection and correction suggestions, alerting users to possible inconsistencies in event definitions; and creating a history of actions, allowing users to review and revert changes if necessary.

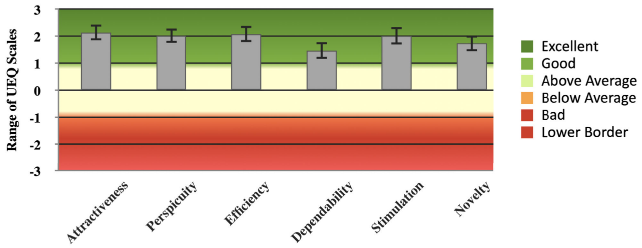


FIGURE 12. Mean value per UEQ scales.

We can group the UEQ scales into *Attractiveness*, and *Pragmatic* and *Hedonic* Quality [50]. Figure 13 shows the mean values per each group. The *Attractiveness* has the highest value indicating that STEVE was considered by the users excellent in its overall impression, attractive, enjoyable, or pleasing. For the other groups, it was also classified as *Excellent* [50] by the users. Therefore, we can evaluate the users' experience in the STEVE experiments as excellent regarding pragmatic and hedonic aspects, achieving our *G5* goal.

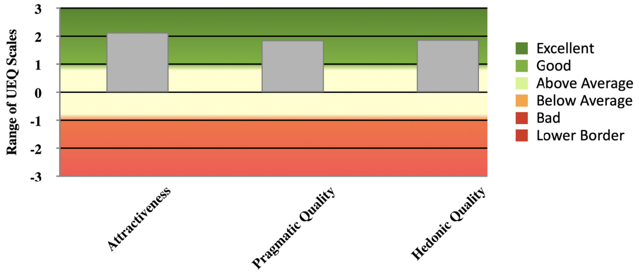


FIGURE 13. Mean value per UEQ group.

C. DISCUSSION

Following our GQM-based evaluation structure, we defined five goals for our experiments. We defined a set of questions for each goal, and for each question, we defined metrics to achieve the goals. We analyzed the effectiveness and perspicuity of STEVE's manual synchronization and temporal and interactivity relations. In addition, we used the SUS and UEQ methodologies to evaluate the STEVE's overall usability and user experience.

To overview the experiment's results, most users successfully used the manual and temporal relation synchronizations and the interactivity feature. However, some users reported they could not understand the interactivity definition, which is according to our G3 analysis. The lower scores for this functionality may be related to the complexity to understand how asynchronous relationships can be specified together with synchronous event-based relations. To mitigate this, some actions can be taken, such as improvements to the STEVE 2.0 interface, simplifying the visual representation of interactions, adding explanatory tooltips to the user interface during interactivity definition; expanded documentation, which would serve as a quick reference guide with practical examples of common interactions (e.g. "How to link a button to a video"); and offering personalized training, with supervised hands-on sessions to explain key concepts of the tool (e.g. "asynchronous events").

From SUS analysis, we evaluated STEVE's overall usability as *Acceptable*/*Good* [52]. Using the UEQ methodology, we achieved our *G5* goal, classifying STEVE user experience as *Excellent* [50] for all UEQ scales, except for *Dependability*, which was evaluated as *Above Average* [50]. That scale indicates that we can improve STEVE to be more consistent in avoiding unexpected behaviors.

Our experiment showed that users without programming experience could successfully create mulsemedia applications using STEVE 2.0. Additionally, defining interactivity in temporal-view authoring tools remains a complex task and requires further investigation in future usability studies

D. LIMITATIONS

We acknowledge that the evaluation of STEVE 2.0 could benefit from a more diverse sample of participants. Nevertheless, we would like to highlight that 6 of the 9 participants from non-technology backgrounds successfully completed all tasks. Although most participants shared similar backgrounds, the large number of responses enabled a preliminary analysis of application use and usability.

In addition, we used consolidated methodologies, such as SUS and UEQ, to ensure a rigorous usability analysis without bias. The results suggest that, despite the predominantly technological sample, the tool demonstrates strong usability and potential for users from different areas.

Regarding remote evaluation, the choice for a remote experiment was motivated by logistical constraints (geographical availability). To reduce errors, we provided tutorials and videos to standardize the experience and manually verified submitted projects to confirm task completion. Furthermore, the data collected via SUS and UEQ show a high degree of usability and positive experience, which suggests that the results are consistent even without direct observation.

Regarding the lack of direct comparisons with similar technologies, it is important to highlight that this has not occurred because, based on our knowledge, there are no tools that operate in the same way as STEVE 2.0. This makes it difficult to measure its differences in relation to other approaches objectively and prevents a quantitative assessment of its efficiency and performance in a competitive context. Furthermore, without a direct reference, it is more challenging to determine which aspects of the tool can be improved based on practices already established in the market.

Possible alternatives for this include conducting an exhaustion test (also known as a load test), to verify the functionality of the application under extreme conditions of use, such as including different media and sensory effects; comparison with video editors, because even with different proposals, the interfaces and modes of use are similar, allowing us to infer strategies and techniques that can be inserted into STEVE 2.0 to improve its workflow and usability; and testing with experienced users of other multimedia authoring tools, collecting comparative feedback on the learning curve, fluidity of the creation process and potential for adoption of the tool.

Despite these challenges, the results obtained are still robust. The study provides a reliable starting point, especially in the evaluation based on the SUS and UEQ, indicating high perspicuity, smooth learning curve, and ease of use, desired features for STEVE 2.0. It is also worth noting that 81.8% of the users completed all tasks successfully, confirming the alignment of the tool with the initial envisioned objective. However, we plan to conduct a new round of supervised testing involving users from different areas, such as designers, educationalists, and industry professionals, to confirm and expand the conclusions obtained so far.

VI. CONCLUSION

This paper proposed an approach for authoring mulsemedia applications based on events that includes a mulsemedia conceptual model (MultiSEM) and a graphical authoring environment for mulsemedia applications named (STEVE 2.0). A conceptual model is essential to represent the spatio-temporal behavior of mulsemedia documents in a structured fashion. Furthermore, conceptual models aid the specification of document nodes (media and sensory effects), providing entities to represent their content and presentation characteristics.

Our analysis focuses on STEVE 2.0, an authoring environment for mulsemedia applications. This tool demonstrates how MultiSEM supports the development of mulsemedia environments through temporal visualizations. Additionally, we examine its usability, key features, and overall user experience, showing that our approach can enhance the authoring phase of such applications. STEVE is integrated with NCL 4.0 and Ginga-NCL, which function as comprehensive mulsemedia platforms, providing end-to-end support across the production, distribution, and rendering stages of mulsemedia applications.

TABLE 3. Questions for G1 goal.

Question	Description
Q1	I found STEVE's sensory effect icons easy to understand.
Q2	I found changing sensory effect duration hard in STEVE.
Q3	I found synchronizing sensory effects with video scenes easy in STEVE.
Q4	Did you successfully complete this task?

TABLE 4. Questions for G2 goal.

Question	Description
Q1'	I found the temporal synchronization between media and sensory effects easy to use.
Q2'	I found the temporal alignments hard to use.
Q3'	I found the temporal alignment icons easy to understand.
Q4'	Which temporal alignments did you use?
Q4	Did you successfully complete this task?

TABLE 5. Questions for G3 goal.

Question	Description
Q1"	I found the interactivity functionality easy to use.
Q4	Did you successfully complete this task?

TABLE 6. SUS questionnaire for STEVE.

Question	Description
Q1	I think that I would like to use STEVE frequently.
Q2	I found STEVE unnecessarily complex.
Q3	I thought STEVE was easy to use.
Q4	I think that I would need the support of a technical person to be able to use STEVE.
Q5	I found the various functions in STEVE were well integrated.
Q6	I thought there was too much inconsistency in STEVE.
Q7	I would imagine that most people would learn to use STEVE very quickly.
Q8	I found STEVE very cumbersome (awkward) to use.
Q9	I felt very confident using STEVE.
Q10	I needed to learn a lot of things before I could get going with STEVE.

A. FUTURE DIRECTIONS

Based on the findings and limitations identified during the usability evaluation of STEVE 2.0, several future directions have been outlined to enhance the tool's capabilities and support its adoption across broader contexts.

One key area of improvement involves the interactivity design feature. Although most participants were able to use it, this functionality received lower usability scores, especially among users unfamiliar with event-based paradigms. Future versions of STEVE will offer more intuitive ways to configure interactivity, including improved visual representations, contextual tooltips, and guided workflows. These enhancements aim to reduce the learning curve and make the creation of asynchronous behaviors more accessible to novice users.

Another important direction involves conducting broader and supervised usability testing. The current study was based on remote participation, which, despite its scale, introduced

TABLE 7. UEQ - user experience questionnaire.

		Description	Opposite Meanings
Attractiveness		Overall impression of the product. Do users like or dislike it? Is it attractive, enjoyable, or pleasing?	annoying / enjoyable bad / good unlikable / pleasing unpleasant / pleasant unattractive / attractive unfriendly / friendly
Pragmatic	**Perspicuity**	Is it easy to get familiar with the product? Is it easy to learn? Is the product easy to understand and clear?	not understandable / understandable easy to learn / difficult to learn complicated / easy clear / confusing
	Efficiency	Can users solve their tasks without unnecessary effort? Is the interaction efficient and fast? Does the product react fast to user input?	fast / slow inefficient / efficient impractical / practical organized / cluttered
	Dependability	Does the user feel in control of the interaction? Can they predict the system behavior? Does the user feel safe when using the product?	unpredictable / predictable obstructive / supportive secure / not secure meets expectations / does not meet expectations
Hedonic	**Stimulation**	Is it exciting and motivating to use the product? Is it fun to use?	valuable / inferior boring / exciting not interesting / interesting motivating / demotivating
	Novelty	Is the product innovative and creative? Does it capture users' attention?	creative / dull inventive / conventional usual / leading-edge conservative / innovative

variability in user experience and support. To address this, future evaluations will be carried out in controlled environments, engaging participants from diverse backgrounds such as education, design, and multimedia production. This approach will help validate the tool's applicability in varied professional contexts and refine its design based on more heterogeneous user profiles.

Although no existing platform offers functionality equivalent to STEVE 2.0, comparative usability studies will be conducted with conventional authoring tools, including video editors and immersive content platforms. These studies will assess aspects such as expressiveness, ease of use, authoring efficiency, and the ability to handle multisensory synchronization, thereby providing external benchmarks to guide the continued development of STEVE.

Future work also includes extending STEVE 2.0 to support 360° video and Virtual Reality content. This integration will enable authors to define and synchronize sensory effects within immersive spherical environments, responding to user orientation, gaze, or movement. Addressing this challenge will require updates to the graphical interface and rendering architecture, allowing for a more immersive authoring and playback experience.

Finally, the adoption of intelligent authoring assistance is expected to significantly streamline the creation process. By incorporating machine learning models, STEVE will be able to suggest sensory effects based on video content, automatically detect inconsistencies in event definitions, and provide real-time correction suggestions. These features will help reduce cognitive load, improve content consistency, and accelerate the authoring workflow.

Together, these future directions aim to establish STEVE 2.0 as a flexible, scalable, and intelligent platform for the design of rich multisensory applications across domains such as education, entertainment, healthcare, and interactive broadcasting.

ACKNOWLEDGMENT

This work was partially supported by Brazilian Coordination for the Improvement of Higher Education Personnel (CAPES), National Council for Scientific and Technological Development (CNPq), Carlos Chagas Filho Foundation for Research Support of the State of Rio de Janeiro (FAPERJ), and Funding Agency for Studies and Projects (FINEP). The authors also express their gratitude to the CAPES PRINT Program for its partial financial support.

The Article Processing Charge for the publication of this research was funded by the Brazilian Coordination for the Improvement of Higher Education Personnel (CAPES) (ROR ID: 00x0ma614). For the purpose of open access, the authors have applied the Creative Commons CC BY license to any accepted version of the article.

APPENDIX

USER EXPERIENCE QUESTIONNAIRES

See Tables 3–7.

REFERENCES

[1] K. Gsöllpointner, R. Schnell, and R. K. Schuler, *Digital Synesthesia: A Model for the Aesthetics of Digital Art*. Berlin, Germany: Walter de Gruyter, 2016.

[2] G. Ghinea, C. Timmerer, W. Lin, and S. R. Gulliver, "Mulsemedia: State of the art, perspectives, and challenges," *ACM Trans. Multimedia Comput., Commun., Appl.*, vol. 11, no. 1s, p. 17, Jul. 2014.

[3] M. Waltl, C. Timmerer, and H. Hellwagner, "Improving the quality of multimedia experience through sensory effects," in *Proc. 2nd Int. Workshop Quality Multimedia Exper. (QoMEX)*, Jun. 2010, pp. 124–129.

[4] B. Rainer, M. Waltl, E. Cheng, M. Shujau, C. Timmerer, S. Davis, I. Burnett, C. Ritz, and H. Hellwagner, "Investigating the impact of sensory effects on the quality of experience and emotional response in web videos," in *Proc. 4th Int. Workshop Quality Multimedia Exper.*, Jul. 2012, pp. 278–283.

[5] Z. Yuan, G. Ghinea, and G.-M. Muntean, "Beyond multimedia adaptation: Quality of experience-aware multi-sensorial media delivery," *IEEE Trans. Multimedia*, vol. 17, no. 1, pp. 104–117, Jan. 2015.

[6] J. Monks, A. Olaru, I. Tal, and G.-M. Muntean, "Quality of experience assessment of 3D video synchronised with multisensorial media components," in *Proc. IEEE Int. Symp. Broadband Multimedia Syst. Broadcast. (BMSB)*, Jun. 2017, pp. 1–6.

[7] A. Covaci, L. Zou, I. Tal, G.-M. Muntean, and G. Ghinea, "Is multimedia multisensorial?—A review of mulsemedia systems," *ACM Comput. Surveys*, vol. 51, no. 5, p. 91, Jul. 2018.

[8] M. N. de Amorim, E. B. Saleme, F. R. de Assis Neto, C. A. Santos, and G. Ghinea, "Crowdsourcing authoring of sensory effects on videos," *Multimedia Tools Appl.*, vol. 78, no. 14, pp. 19201–19227, Jul. 2019.

[9] Á. L. V. Guedes, R. G. D. A. Azevedo, and S. D. J. Barbosa, "Extending multimedia languages to support multimodal user interactions," *Multimedia Tools Appl.*, vol. 76, no. 4, pp. 5691–5720, Feb. 2017.

[10] C. A. S. Santos, A. N. R. Neto, and E. B. Saleme, "An event driven approach for integrating multi-sensory effects to interactive environments," in *Proc. IEEE Int. Conf. Syst., Man, Cybern.*, Oct. 2015, pp. 981–986.

[11] E. B. Saleme, A. Covaci, G. Mesfin, C. A. S. Santos, and G. Ghinea, "Mulsemedia DIY: A survey of devices and a tutorial for building your own mulsemedia environment," *ACM Comput. Surv.*, vol. 52, no. 3, pp. 1–29, May 2020.

[12] L. F. G. Soares, R. F. Rodrigues, and D. C. M. Saade, "Modeling, authoring and formatting hypermedia documents in the HyperProp system," *Multimedia Syst.*, vol. 8, no. 2, pp. 118–134, Mar. 2000.

[13] H. L. Hardman, "Modelling and authoring hypermedia documents," Ph.D. thesis, Dept. Comput. Sci., Univ. Amsterdam, Amsterdam, The Netherlands, 1998. [Online]. Available: http://www.cwi.nl/ lynda/thesis/

[14] S. Kim and J. Han. (2004). *Text White Paper MPEG-V*. ISO. [Online]. Available: https://www.mpeg.org/whitepapers/

[15] M. Waltl, B. Rainer, C. Timmerer, and H. Hellwagner, "An end-to-end tool chain for sensory experience based on MPEG-V," *Signal Process., Image Commun.*, vol. 28, no. 2, pp. 136–150, Feb. 2013.

[16] B. Choi, E.-S. Lee, and K. Yoon, "Streaming media with sensory effect," in *Proc. Int. Conf. Inf. Sci. Appl.*, Apr. 2011, pp. 1–6.

[17] S.-K. Kim, "Authoring multisensorial content," *Signal Process., Image Commun.*, vol. 28, no. 2, pp. 162–167, Feb. 2013.

[18] G. Blakowski and R. Steinmetz, "A media synchronization survey: Reference model, specification, and case studies," *IEEE J. Sel. Areas Commun.*, vol. 14, no. 1, pp. 5–35, Jun. 1996.

[19] M. F. de Sousa, C. A. G. Ferraz, R. Kulesza, I. Ayres, and M. Lima, "MulSeMaker: An MDD tool for MulSeMedia web application development," in *Proc. 23rd Brazillian Symp. Multimedia Web*, Oct. 2017, pp. 317–324.

[20] D. C. Muchaluat-Saade and L. F. G. Soares, "XConnector and XTemplate: Improving the expressiveness and reuse in web authoring languages," *New Rev. Hypermedia Multimedia*, vol. 8, no. 1, pp. 139–169, Jan. 2002.

[21] J. F. Allen, "Maintaining knowledge about temporal intervals," *Commun. ACM*, vol. 26, no. 11, pp. 832–843, Nov. 1983.

[22] M. Josué, R. Abreu, F. Barreto, D. Mattos, G. Amorim, J. dos Santos, and D. Muchaluat-Saade, "Modeling sensory effects as first-class entities in multimedia applications," in *Proc. 9th ACM Multimedia Syst. Conf.*, Jun. 2018, pp. 225–236.

[23] D. P. D. Mattos and D. C. M. Saade, "STEVE: Spatial–temporal view editor for authoring hypermedia documents," in *Proc. 22nd Brazilian Symp. Multimedia Web*, Nov. 2016, pp. 63–70.

[24] D. P. de Mattos, D. C. Muchaluat-Saade, and G. Ghinea, "An approach for authoring mulsemedia documents based on events," in *Proc. Int. Conf. Comput., Netw. Commun. (ICNC)*, Feb. 2020, pp. 273–277.

[25] R. S. de Abreu, D. Mattos, J. de Santos, G. Ghinea, and D. C. Muchaluat-Saade, "Toward content-driven intelligent authoring of mulsemedia applications," *IEEE MultimediaMag.*, vol. 28, no. 1, pp. 7–16, Jan. 2021.

[26] I.-T. Rec. (2009). *H. 761, Nested Context Language (NCL) and GINGA-NCL for IPTV Services, Geneva, Apr. 2009*. [Online]. Available: https://www.itu.int/rec/T-REC-H.761

[27] M. Josué, M. Moreno, and D. Muchaluat-Saade, "Mulsemedia preparation: A new event type for preparing media object presentation and sensory effect rendering," in *Proc. 10th ACM Multimedia Syst. Conf.*, Jun. 2019, pp. 110–120.

[28] D. P. D. Mattos, D. C. Muchaluat-Saade, and G. Ghinea, "Beyond multimedia authoring: On the need for mulsemedia authoring tools," *ACM Comput. Surv.*, vol. 54, no. 7, pp. 1–31, Sep. 2022.

[29] F. Barreto, R. S. de Abreu, M. I. P. Josué, E. B. B. Montevecchi, P. A. Valentim, and D. C. Muchaluat-Saade, "Providing multimodal and multi-user interactions for digital TV applications," *Multimedia Tools Appl.*, vol. 82, no. 4, pp. 4821–4846, Feb. 2023.

[30] Y. Sulema and V. Glinskii, "Semantics and pragmatics of programming language asampl," *Problems Program.*, vol. 1, no. 1, pp. 74–83, 2020.

[31] V. Y. Peschanskii, "Timewise data processing with programming language asampl," *Sci. Notes Taurida Nat. VI. Vernadsky Univ. Serires, Tech. Sci.*, vol. 1, no. 1, pp. 132–137, 2020.

[32] Y. Sulema, "Asampl: Programming language for mulsemedia data processing based on algebraic system of aggregates," in *Interactive Mobile Communication, Technologies and Learning*. Cham, Switzerland: Springer, 2017, pp. 431–442.

[33] V. Girotto, E. Walker, and W. Burleson, "CrowdMuse: Supporting crowd idea generation through user modeling and adaptation," in *Proc. Creativity Cognition*, Jun. 2019, pp. 95–106.

[34] F. Danieau, J. Bernon, J. Fleureau, P. Guillotel, N. Mollet, M. Christie, and A. Lécuyer, "H-studio: An authoring tool for adding haptic and motion effects to audiovisual content," in *Proc. 26th Annu. ACM Symp. User Interface Softw. Technol. (UIST)*, Jul. 2013, pp. 83–84.

[35] S.-H. Shin, K.-S. Ha, H.-O. Yun, and Y.-S. Nam, "Realistic media authoring tool based on MPEG-V international standard," in *Proc. 8th Int. Conf. Ubiquitous Future Netw. (ICUFN)*, Jul. 2016, pp. 730–732.

[36] C. Jost and B. Le Pévédic, "How to integrate interactions into video editing software?" in *Proc. 2nd Workshop Multisensory Exper. (SensoryX)*, Jun. 2022, pp. 19–24.

[37] L. Jalal and M. Murroni, "Enhancing TV broadcasting services: A survey on mulsemedia quality of experience," in *Proc. IEEE Int. Symp. Broadband Multimedia Syst. Broadcast. (BMSB)*, Jun. 2017, pp. 1–7.

[38] H. Coelho, M. Melo, L. Barbosa, J. Martins, M. S. Teixeira, and M. Bessa, "Authoring tools for creating 360 multisensory videos—Evaluation of different interfaces," *Expert Syst.*, vol. 38, no. 5, p. 12418, Aug. 2021.

[39] H. Coelho, M. Melo, J. Martins, and M. Bessa, "Collaborative immersive authoring tool for real-time creation of multisensory VR experiences," *Multimedia Tools Appl.*, vol. 78, no. 14, pp. 19473–19493, Jul. 2019.

[40] F. M. D. Farias, D. P. De Mattos, G. Ghinea, and D. C. Muchaluat-Saade, "Immersive authoring of 360 degree interactive applications," *IEEE Access*, vol. 10, pp. 115205–115221, 2022.

[41] R. Abreu, D. Mattos, J. Santos, G. Guinea, and D. C. Muchaluat-Saade, "Semi-automatic mulsemedia authoring analysis from the user's perspective," in *Proc. 14th ACM Multimedia Syst. Conf.* New York, NY, USA: ACM, Jun. 2023, pp. 249–256.

[42] S.-K. Kim, S.-J. Yang, C. H. Ahn, and Y. S. Joo, "Sensorial information extraction and mapping to generate temperature sensory effects," *ETRI J.*, vol. 36, no. 2, pp. 224–231, Apr. 2014.

[43] D. P. de Mattos and D. C. Muchaluat-Saade, "STEVE: A hypermedia authoring tool based on the simple interactive multimedia model," in *Proc. ACM Symp. Document Eng.*, Aug. 2018, pp. 1–10.

[44] J. A. F. dos Santos, J. V. Silva, R. R. Vasconcelos, W. Schau, C. Werner, and D. C. Muchaluat-Saade, "ANA: API for NCL authoring," in *Proc. 18th Brazilian Symp. Multimedia Web—Workshop Tools Appl.*, Jul. 2012, pp. 63–66.

[45] R. M. D. R. Costa, M. F. Moreno, and L. F. Gomes Soares, "Intermedia synchronization management in DTV systems," in *Proc. 8th ACM Symp. Document Eng.*, Sep. 2008, pp. 289–297.

[46] E. C. O. Silva, J. A. F. dos Santos, and D. C. Muchaluat-Saade, "NCL4WEB: Translating NCL applications to HTML5 web pages," in *Proc. ACM Symp. Document Eng.*, Sep. 2013, pp. 253–262.

[47] B. Liu, W. Wang, Y. Zhang, R. Huang, and J. Raiti, "Lullaland: A multisensory virtual reality experience to reduce stress," in *Proc. Extended Abstr. CHI Conf. Hum. Factors Comput. Syst.* New York, NY, USA: ACM, Apr. 2023, pp. 1–6, doi: 10.1145/3544549.3585636.

[48] V. Basili, G. Caldiera, F. McGarry, R. Pajerski, G. Page, and S. Waligora, "The software engineering laboratory–an operational software experience factory," in *Proc. Int. Conf. Softw. Eng.*, 1992, pp. 370–381.

[49] J. Brooke, "SUS-A quick and dirty usability scale," *Usability Eval. Ind.*, vol. 189, no. 194, pp. 4–7, 1996.

[50] M. Schrepp, A. Hinderks, and J. Thomaschewski, "Design and evaluation of a short version of the user experience questionnaire (UEQ-S)," *Int. J. Interact. Multimedia Artif. Intell.*, vol. 4, no. 6, pp. 103–108, 2017.

[51] J. Sauro and J. R. Lewis, *Quantifying the User Experience: Practical Statistics for User Research*. San Mateo, CA, USA: Morgan Kaufmann, 2016.

[52] A. Bangor, P. T. Kortum, and J. T. Miller, "An empirical evaluation of the system usability scale," *Int. J. Hum.-Comput. Interact.*, vol. 24, no. 6, pp. 574–594, Jul. 2008.

DOUGLAS MATTOS received the degree (Hons.), the master's degree in computer science, and the Ph.D. degree in computer science in mulsemedia and multimedia systems from Fluminense Federal University (UFF), in 2012, 2016, and 2021, respectively, where he proposed an approach to flacilitate the authoring of multimedia applications with multiple sensory effects. He developed new technologies I facilitate the authoring of hypermedia applications for digital TV and the web with UFF. He was a Visiting Researcher during his internship period with the Brunel University of London, in 2019. As a Researcher and a Software Developer with the MídiaCom Laboratory, he participated in scientific projects in the areas of teleprotection based on the IEC-61850 standard (in partnership with TAESA, 2020) and edge computing and virtual network functions (in partnership with DELL, 2021). He was a Tutor and a Final Project Advisor in the Technology in Computer Systems Program at CEDERJ, from 2016 to 2021. He was a Researcher in the Erasmus VR4STEM Project at Politehnica University of Bucharest, in 2017. Additionally, he taught undergraduate courses in computer science with Infnet, from 2018 to 2019, focusing on Java programming. In the industry, he has been a Software Engineer specializing in Java technologies, since 2013.

RÔMULO VIEIRA (Member, IEEE) received the bachelor's degree in control and automation engineering from President Antonio Carlos University (UNIPAC), Conselheiro Lafaiete, Minas Gerais, in 2019, and the master's degree in computer science from the Federal University of São João del-Rei (UFSJ), in 2021. He is currently pursuing the Ph.D. degree in computer science with Fluminense Federal University (UFF), with an internship period at Centrum Wiskunde en Informatica (CWI), The Netherlands. He was part of the Research and Development Team for Brazilian TV 3.0 standard, working on the development of applications for virtual reality (VR) support and multi-device integration. He has experience in the Musical Internet of Things, multimedia systems, computer music, human–computer interaction/UX design, web development, and computer networks. As a Multimedia Artist, he is a Music Producer, a Photographer, and a Video Editor, and in areas related to computer music, electronic art, and interactivity.

DÉBORA C. MUCHALUAT-SAADE (Member, IEEE) received the Ph.D. degree in computer science from the Pontifical Catholic University of Rio de Janeiro (PUC-Rio), in 2003. She is currently a Full Professor with the Department of Computer Science, Fluminense Federal University (UFF), Brazil. She is one of the founders and heads of the MídiaCom Research Laboratory. She is a CNPq and FAPERJ Fellow. She participated in the development of the NCL language-nested context language-adopted as the ABNT NBR 15606-2 standard in the GINGA middleware of the Brazilian Digital TV System and as an international recommendation by ITU-T H.761 for IPTV services. Her main research interests include multimedia systems, computer networks, the IoT, smart grids, socially assistive robots, and digital health. She is a member of the Council of Brazilian Computing Society (SBC) and of Brazilian Digital TV System Forum (Forum SBTVD).

GEORGE GHINEA (Member, IEEE) received the B.Sc. degree with majors in computer science and mathematics and the B.Sc. (Hons.) and M.Sc. degrees in computer science from the University of the Witwatersrand, South Africa, and the Ph.D. degree in quality of perception: an essential facet of multimedia communications from the University of Reading, U.K., in 2000. He is currently a Professor with the Department of Computer Science, Brunel University of London. In his doctoral research, he proposed the quality of perception (QoP) metric, a precursor to the now widely recognized quality of experience (QoE) concept. However, while QoE remains a conceptual framework, QoP serves as a Concrete Metric. He recognizing the infotainment duality of multimedia, QoP not only quantifies the subjective enjoyment derived from multimedia experiences but also assesses how effectively such presentations aid in the assimilation of informational content. His research activities lie at the intersection of computer science, media, and psychology. His work primarily focuses on perceptual multimedia quality and the development of end-to-end communication systems that integrate user perceptual requirements. His expertise has been applied in diverse areas, including eye-tracking, telemedicine, multimodal interaction, and ubiquitous and mobile computing. His research has been funded by both national and international funding bodies, all of it resulting from collaborative efforts with various teams and stakeholders he has had the privilege to work with. Over his career, he has supervised 33 Ph.D. students to completion and has co-authored over 350 high-quality research publications with his students and research collaborators.

Coordenação de Aperfeiçoamento de Pessoal de Nível Superior (CAPES) - ROR identifier: 00x0ma614