

Review

Not peer-reviewed version

Towards Sustainable Virtual Reality: Gathering Design Guidelines for Intuitive Authoring Tools

[Iolanda Lima Chamusca](#)*, [Cristiano Vasconcellos Ferreira](#), [Thiago Barros Murari](#),
[Antonio Lopes Apolinário Jr.](#), [Ingrid Winkler](#)*

Posted Date: 17 January 2023

doi: 10.20944/preprints202301.0291.v1

Keywords: virtual reality; authoring tools; intuitiveness; digitization; sustainability; user-centered design; human-computer interaction; metaverse



Preprints.org is a free multidiscipline platform providing preprint service that is dedicated to making early versions of research outputs permanently available and citable. Preprints posted at Preprints.org appear in Web of Science, Crossref, Google Scholar, Scilit, Europe PMC.

Copyright: This is an open access article distributed under the Creative Commons Attribution License which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Review

Towards Sustainable Virtual Reality: Gathering Design Guidelines for Intuitive Authoring Tools

Iolanda Lima Chamusca ^{1,*} , Cristiano Vasconcellos Ferreira ² , Thiago Barros Murari ³ , Antonio Lopes Apolinario Jr. ⁴  and Ingrid Winkler ^{3,*} 

¹ Ford Motor Company, Camaçari 42831710, Brazil

² Technological Center of Joinville, Federal University of Santa Catarina, Rua Dona Francisca, 8300, Zona Industrial Norte, Joinville, SC 89219-600, Brazil; cristiano.v.ferreira@ufsc.br

³ Department of Management and Industrial Technology, SENAI CIMATEC University Center, Salvador 41650-010, Brazil; thiago.murari@fieb.org.br

⁴ Department of Computer Science, Federal University of Bahia, Salvador 40170110, Brazil; antonio.apolinario@ufba.br

* Correspondence: ichamusc@ford.com (I.L.C.); ingrid.winkler@doc.senaicimatec.edu.br (I.W.)

Abstract: Virtual reality experiences are frequently created using game engines, yet they are not simple for novices and unskilled professionals who don't have programming and 3D modeling skills. Concurrently, there is a knowledge gap in software project design for intuitive virtual reality authoring tools, which were supposed to be easier to use. This work compiles design guidelines derived from a systematic literature review to contribute in the development of more intuitive virtual reality authoring tools. We searched the Scopus and Web of Science databases for works published between 2018 and 2021 and discovered 14 papers. We compiled 14 requirement and feature design guidelines, such as Visual Programming, Immersive Authoring, Reutilization, Sharing and Collaboration, Metaphors, and Movement Freedom, among others. The collected guidelines have the potential to either guide the development of new authoring tools or to evaluate the intuitiveness of existing tools. Furthermore, they can also support the development of the metaverse, since virtual content creation is one of its bases.

Keywords: virtual reality; authoring tools; intuitiveness; digitization; sustainability; user-centered design; human-computer interaction; metaverse

1. Introduction

Organizations worldwide have been conducting a variety of initiatives to explore new technological solutions and to exploit their benefits, including transformations of business operations, products, processes, structures and management concepts [1]. Digital Transformation is about adopting these disruptive technological solutions, like Virtual Reality (VR), to increase productivity, changing how people interact with it, either as consumers or professionals [2].

VR has helped organizations to achieve several Sustainable Development Goals (SGD). Immersive experiences improve education, raise citizen awareness, and support behavior change toward more sustainable choices, from plastic pollution to building design, sustainable mobility, tourism, and water management [3], contributing to achieve SDG 5 (Gender Equality), SDG 11 (Sustainable Cities and Communities), SDG 12 (Responsible Consumption and Production), and SDG 13 (Climate Accuracy). In product development, VR has been essential to a more sustainable process, reducing carbon emissions by replacing physical products or real-world interactions with virtual ones [4] and foreseeing ergonomic risks, protecting workers from risk of harm and improving workplace well-being [5]. As a result, it is a critical contribution to achieving SDG 8 (Decent work for all) and SDG 9 (Industry, Innovation, and Infrastructure), which calls for more sustainable industrialization, resource efficiency, and clean, environmentally sound technology and industrial processes. Therefore, a more sustainable VR contributes directly to a more sustainable future.

On the other hand, creating virtual reality experiences isn't widespread and requires expensive and lengthy development processes using game engines such as Unreal¹ and Unity², which demand expert professionals [6–8]. This is due to the complexity of the software architecture of virtual reality systems, which involves a great diversity of resources including unusual input and output devices such as Head-Mounted Displays (HMD), tracking systems, 3D mice, and others [9]. In addition, for a true digital transformation to take place in the economy and society we live in, it is necessary to ensure that more people are able not only to use VR experiences but also have the skill to create them.

This complex nature of immersive technology requires a multidisciplinary profile from professionals, which includes considerable technical knowledge in programming languages and/or 3D modeling [8,10–12]. Therefore, developing interactive scenes in VR is challenging and uncomfortable for novices, which include people who come from high level creator groups, such as digital artists and designers, and developers who come from other technology fields, instead of immersive [11,12]. Beginners also include professionals such as teachers, doctors, engineers, and other professionals who only want to use VR as a supplement to their day-to-day work [13–15].

An alternative to the long learning curve is to adopt authoring tools, as they aim to enable an efficient content creation through minimal changes. The term "authoring tool" refers to software structures that include the most important tools and features of content creation while making product maintenance faster and better [16,17]. These tools are used for low-fidelity authoring, which requires less programming skills, as opposed to high-fidelity prototyping, which requires advanced programming skills [11].

There are, currently, a range of authoring tools for creating virtual reality experiences, with many of them available as open source software [18]. These tools, however, are frequently limited not only in functionality but also in documentation and tutorials, making them unsuitable for supporting the entire development cycle [11,19]. This occurs because these authoring tools are often developed as proof-of-concept, to help test the user's acceptance and identify main features they might enjoy to see on the final product, but not with an application field in mind [16].

These factors contribute to the technology's lack of maturity, leading to the difficulty of defining best practices for multiple user applications, the lack of standardized processes, the lack of recommended practices, the lack of a common language, and the lack of interoperability between VR tools and between pre-existing data such as 3D assets and codes [6,11,14,17]. Combined with the complexity of virtual reality systems architecture, these barriers contribute to the persistence of novices' learning difficulties, requiring good prior knowledge to utilize the full potential of the tool [12].

Previous works have addressed various aspects of using authoring tools for the creation of VR experiences ([11,16,20]). These studies complement one another: while [20] introduced eight key barriers to creating VR/AR experiences, [11] investigated the challenges that VR creators face when using authoring tools for collaborative teamwork, and [16] systematically reviewed studies on authoring tool development to analyze their usability evaluation methods.

These works have concentrated on analyzing how authoring tools were used after they were developed and made available to users rather than on how the development process affected the final product. Therefore, there's a knowledge gap regarding the design of authoring tool software, which we seek to mitigate by listing design guidelines to assist software developers during the project defining phase. The guidelines will help them choose and create the features and requirements that these tools must have to be considered intuitive [21].

Effective human-machine communication is needed to understand probable interactions, what's happening in the present, and what can happen next [22]. Human-centered design prioritizes human needs, skills, and behaviors, then designs to meet them. With the design guidelines described in [22],

¹ <https://unity.com/pt>

² <https://www.unrealengine.com/en-US>

people can understand how to create well-designed VR experiences, similarly, the guidelines of this work supports the development of intuitive authoring tools to create the VR experience, but not the experience itself. Our results will contribute to structure the discussion about how beginners can accomplish their goals with a more user-friendly and inclusive authoring tool, in a more sustainable VR.

The faster expansion of immersive technology can also bring a huge advantage to concepts such as the metaverse, a major digital transformation that has already been impacting the work format in the technology area, encouraging the need for vacancies for people specialized in using the metaverse in conjunction with a company's strategy. In the metaverse, users can seamlessly experience a digital life as well as make digital creations supported by the metaverse engine, particularly with the assistance of extended reality and human-computer interaction [23]. The virtual worlds that will compose the metaverse need to be created. For that, people and authoring tools will be required to enable the creation of new digital content, and the more people collaborating to build these worlds, the bigger and more diverse it will be.

Thus, the goal of this study is to compile design guidelines derived from a systematic literature review to contribute in the development of more intuitive virtual reality authoring tools. This document is organized as follows: Section 2 describes the materials and methods utilized, Section 3 presents and analyzes the results, and Section 4 provides our conclusions and suggestions for further research.

2. Materials and Methods

This review followed the Preferred Reporting Items for Systematic Reviews and Meta-Analyses (PRISMA) guidelines, which was designed to “help systematic reviewers transparently report why the review was done, what the authors did, and what they found” [24]. Additionally, the process provided in [25], which consists of seven steps, was utilized: planning, defining the scope, searching the published research, assessing the evidence base, synthesizing, analyzing, and writing. This Systematic Literature Review is registered on Open Science Framework, number <https://osf.io/u3q7m>.

As preceded by [4], an expert in virtual reality-based design testing of product development defined the initial search strategy, which was then assessed by three senior researchers individually. The senior researchers had a background in virtual reality applications for industrial innovation, computer graphics, and innovative product development. The strategy resulting from this validation process is described in the sections that follow.

2.1. Planning

The knowledge bases that will be investigated are determined during the planning step [25]. The investigation was carried out using the scientific databases Scopus and Web of Science. These databases were chosen because they are reliable, multi-disciplinary scientific databases of international scope with comprehensive coverage of citation indexing, providing the best data from scientific publications. Scopus now includes 87 million curated documents [26], whereas the Web of Science covers more than 82 million entries [27].

2.2. Defining the Scope

Defining the scope means presenting proper research questions, therefore, three main questions were selected for this systematic review:

Q1: What are the characteristics of the virtual reality authoring tools reviewed?

Q2: What is the definition of intuitiveness in virtual reality authoring tools?

Q3: What are the guidelines for designing intuitive virtual reality authoring tools?

2.3. Literature Search

In the literature search step, a particular string is used to search the database set up in the planning step, based on the research questions asked in the defining the scope step [25]. The first keywords

utilized were “VIRTUAL REALITY” and “VR” to specify the search for this specific cut from immersive technology, therefore not considering augmented reality, with the goal of defining, at the end of this study, design guidelines more focused on VR experiences. The exploratory search was based on articles focused on describing the process to develop “AUTHORING TOOLS”, specifically oriented to create VR experiences, followed by complementary keywords, “SYSTEM” and “FRAMEWORKS”, added to the string to cover other types of software tools.

Because VR experiences can be difficult for novice users to create, the focus is on keywords linked to intuitiveness. From the word “INTUITIVE”, other synonyms were gathered: “FLEXIBLE”, “DEMOCRATIZE”, “ADAPTABLE”, “USABLE”, “FACILITATE”, “SIMPLIFY”, “EASY” and “USER-FRIENDLY”. Thus was formed the final search phrase: TITLE-ABS (("virtual reality" OR "VR") AND ("authoring tools" OR "system" OR "framework") AND ("intuitive" OR "flexible" OR "democratize" OR "adaptable" OR "usable" OR "facilitate" OR "simplify" OR "easy" OR "user-friendly"))).

2.4. Assessing the evidence base

The assessing step uses inclusion and exclusion criteria filters to reduce the number of documents found in the searching the literature step—selecting those that are relevant to the research questions [25]. These criteria were applied to the researched articles in three different phases, as follows:

Phase 1: exclusions through filter options provided by the database used on the research.

- E1.1.: The entry title or abstract didn't have one or more of the terms described on the search phrase;
- E1.2.: Published before 2018;
- E1.3.: Entry not written in English language;
- E1.4.: Virtual reality is not a keyword;
- E1.5.: Duplicate entry.

Phase 2: exclusions through screening of the abstract of publications.

- E2.1.: Entry is theoretical work (e.g., information system proposal, literature review, poster);
- E2.2.: Entry does not consider the development of authoring tools for virtual reality immersive experiences creation;
- E2.3.: Entry focus on augmented reality;
- E2.4.: Entry develops authoring tools for VR experiences creation not based on the use of HMD on virtual environments (e.g., CAVE, 360 video).

Phase 3: exclusion through screening of the entire article, using the tool Mendeley Desktop for organize and classify the publications.

- E3.1.: Entry with less than 5 pages;
- E3.2.: Entry related to the development of authoring tools not directly defined as intuitive and easy-to-use for beginners and unskilled professionals;
- E3.3.: Entry limited on the development of authoring tools specific for an area of application (e.g., health, engineering, education, culture).

The research period chosen was between 2018 and 2021, the last four years before the development of this study, a period equivalent to the fast progression of this technology through these years, not only academically but also on VR hardware releases. For example, the advent of technologically advanced virtual reality headsets in 2016 represented a breakthrough for virtual reality applications and practitioners with the release of the HTC Vive Steam VR headset, the first commercial release of sensor-based tracking [28,29]. Another significant event in the evolution of virtual reality headsets occurred in 2018: Oculus launched the Oculus Go, the first commercially available wireless VR headset with an affordable built-in screen [30].

2.5. Synthesizing and Analyzing

At this point, a professional with experience in the use of virtual reality looked over each document to apply phase 3 of the above exclusion criteria. Figure 1 depicts the flow of the systematic review from searching the published research to synthesizing processes.

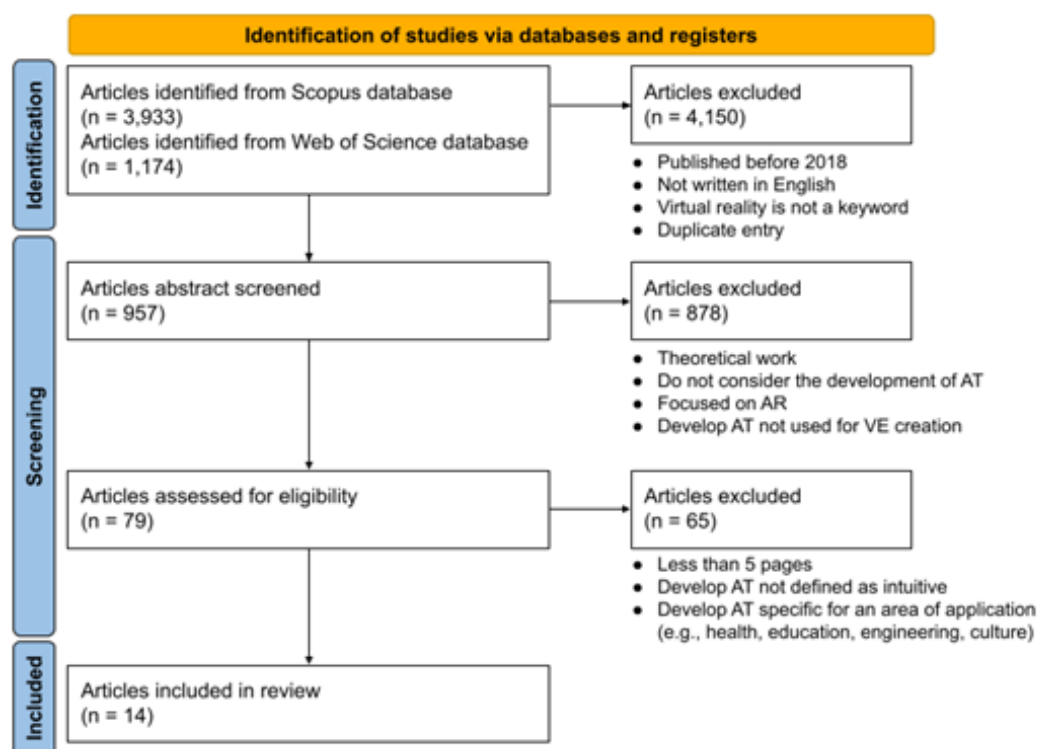


Figure 1. Systematic review flow diagram, adapted from PRISMA [24].

Mendeley Desktop and Microsoft Excel were used to carry out a more detailed analysis of each of the 14 articles. Initially, all the articles were read, and some variables were used as a first qualitative analysis, making it easier to comprehend the later design guidelines presented and what motivated the choice for the 14 articles used in this systematic review. Inspired by the concepts used by [16], in the Table 1 the authoring tools were defined according to:

- **Artifact:** brief description of the authoring tool purpose, it's the artifact developed by the article;
- **Development process (DP):** brief description of the software and hardware tools used in the development and evaluation process of the tool;
- **Type:** "this variable can be classified as two options: plugin or standalone. A plugin is software developed to work over other software to facilitate processes. On the other hand, a standalone is a software that works without any other software and is designed specifically for a purpose" [16].

Then, a list was made with the main concepts pointed out by the authoring tools, which were grouped into requirements and features based on the author's qualitative understanding, a method inspired by the agile methodologies artifacts [31]. The product backlog, one of the artifacts produced by the Scrum, is a prioritized list of product requirements or features that provide business value for the customer [21]. In this work, we considered the virtual reality authoring tools as "products", and the non-expert users as "customers". Below are definitions of the classification terms:

- **Requirement:** generic concepts that delimited the major characteristics of a product, in a project the requirements are obtained by understanding the users needs before the development starts and will help define the product's features;

- Feature: are the “tools” you use within a system to complete a set of tasks or actions, the functionality of a feature will provide the user a desired outcome.

This led to the list of design guidelines presented in results and discussions.

3. Results

In the following sections, the research questions Q1, Q2 and Q3 are addressed.

3.1. Characteristics of the virtual reality authoring tools

The 14 works were reviewed to analyze the research question “Q1: What are the characteristics of the virtual reality authoring tools reviewed?”, and the results are presented in this section. The following is a summary of the main characteristics of the virtual reality authoring tools developed by the reviewed studies:

1. Virtual Environment (VE) creation, where everything that the user sees is a 3D model, also containing collaborative interaction, visual programming and immersive authoring [16];
2. Generic purpose, not developed for the use in an specific field, such as Mechanical Engineering or Medicine [16];
3. Manipulating and importing 3D objects by searching online, either by text or with an immersive sketch in VR mode, editing assets, and adding behaviors;
4. Facilitating interaction between software and hardware through haptic feedback visualization and multisensory stimuli;
5. Interactive human characters development, giving the user pre-setted behaviors such as mouth movements to speak;
6. Artificial intelligence automation using different types of networks to help the user achieve their goals with more efficiency.

Item "1" also represents an exclusion criteria described in section 2.4., which was defined by the fact that most of the authoring tools developed in the field today are aimed at creating 360° videos, as attested by [16]. The 360° experiences allow users to look around freely and are very simple to create, only requiring the application of one or more video files in a VR context to work, nonetheless valuable information can be lost. All the potential interactivity with elements in virtual reality is wasted in a 360° experience, that's why it's a major limitation that the authoring tool only supports 360° videos. In addition, as the authoring tools for creating virtual environments are more complex to use, the present study will have greater value in its contribution towards the intuitiveness of creating VR experiences in virtual environments.

Examining Table 1, the “standalone” authoring tools on the reviewed works were always the web-based ones, using A-FRAME to be built, which is easier to use for proof-of-concept platforms, while the “plugin” authoring tools were always based on Unity3D Engine, a mainstream non-paid game engine. This shows that the authoring tools (standalone and plugins) made in the reviewed works are not ready for the market yet because they have not been released as final products. Table 1 summarizes the variables described in each article:

Table 1. Characteristics of the virtual reality authoring tools.

Ref	Article	Hardware (DP)	Software (DP)	Type
[32]	Tool for 3D assets search through immersive handmaid sketch in VR	Oculus Consumer Version 1 (HMD) and Oculus Touch Controllers	Unity3D Engine and Convolutional Neural Network (CNN)	Plugin
[33]	Tool for visual feedback of haptic properties in VR	HTC Vive Kit (HMD)	Unity3D Engine	Plugin
[34]	Collaborative web-based authoring tool for creating virtual environments in VR	Oculus Rift Consumer version and HTC Vive (HMD)	Node.js / EasyRTC, Socket.IO, WebRTC and A-FRAME (Three.js + WebVR)	Standalone
[35]	Architecture for a collaborative immersive tool to multisensorial experiences creation in VR	HTC Vive (HMD), Bose QuietComfort 25 headphones, Sensory Co SmX-4D for Smell, Buttkickers LFE kit and wind simulator	Unity3D Engine	Plugin
[36]	Immersive authoring tool that allows applying reaction behaviors to objects using visual programming	Oculus Rift (HMD)	NOT INFORMED	Standalone
[37]	Tool for 3D assets editing and application of behaviors to objects	HTC Vive Pro Eye (HMD) and controllers	Unity3D Engine (C#) and SteamVR	Plugin
[38]	Authoring tool for developing interactive agents for VR applications	NOT INFORMED	Unity3D Engine and Cortana for speech recognition	Plugin
[8]	Immersive authoring tool that allows applying reaction behaviors to objects using visual programming	NOT INFORMED	A-FRAME (Three.js and WebVR), Node.js and RxJS	Standalone

Table 1. Cont.

Ref	Article	Hardware (DP)	Software (DP)	Type
[17]	Immersive authoring tool with visual programming to reproduce gamified training scenarios through modular architecture	NOT INFORMED	Unity3D Engine and CodeDOM (.NET Framework)	Plugin
[39]	Authoring tool that uses “nugget tiles” (blocks) so authors can create reduced learning experiences in VR	HTC Vive (HMD)	Unity3D Engine and Virtual Reality Tool-kit (VRTK)	Plugin
[10]	Platform for interactive virtual objects creation in VR through physical objects presented in real world, using AI	Oculus Quest (HMD), Oculus Link, Touch controllers and iPad Pro	Unity3D Engine and AppScanner	Plugin
[40]	Authoring tool that allows flexible control of work spaces for data analysis during collaborative activities in group inside a immersive space in VR	HTC Vive Pro (HMD), Backpack	Unity3D Engine, Immersive Analytics Toolkit (IATK) and Oculus Avatar SDK (body)	Plugin
[41]	Web-based authoring tool with better graphic quality	NOT INFORMED	Unity3D Engine and WordPress / MySQL	Plugin
[12]	Web-based intuitive authoring tool for interactive 3D scenes creation in VR	NOT INFORMED	A-FRAME (Three.js), React and JavaScript	Standalone

3.2. Definition of intuitiveness in virtual reality authoring tools

The 14 studies were examined in relation to the research question “Q2: What is the definition of intuitiveness in virtual reality authoring tools?”. The reviewed works stand out “intuitiveness” as “easy-to-use”, “quickly”, “high usability”, “for non-experts”, “short training”, “simple”, “facilitate” and “reduce complexity”. In these studies, intuitiveness is related to completing tasks quickly, requiring minimal learning, lowering the entry barrier, reducing information, time, and steps, being appropriate for both expert and non-expert users, being aware of and feeling present in virtual reality, feeling comfortable with the tool, making few mistakes, and using natural movements in virtual reality.

It's not possible to evaluate or measure intuitiveness, but we may measure it with usability, effectiveness, efficiency, and satisfaction, using well-established questionnaires and methods (some of them listed in the table below). For example, usability can be measured with System Usability Scale (SUS), effectiveness can be measured by tasks completed successfully, number of errors, and number of help requests, efficiency can be measured by time spent to complete a task and perceived workload, and satisfaction can be measured with the After-Scenario Questionnaire (ASQ) [16,42]. Other measures can be taken into account, such as learnability (time to learn a tool) and recommendation to others.

High-technology products need to exhibit good usability, a qualitative measure of the ease with which a human can employ the functions and features offered [21]. In this work, “intuitive” refers to the quality of an easy-to-use authoring tool whose usability, effectiveness, efficiency, and satisfaction evaluation showed positive results.

In terms of usability evaluation methods, nine studies adopted Likert-scale surveys [8,10,12,17,32,33,37,40,41], three used SUS [10,12,35], four adopted other types of questionnaires like ASQ and NASA TLX [35,37,39,41], three used qualitative retrospective interviews [8,10,36], two implemented other methods like the “thinking-aloud method” or measured the number of errors and time taken to complete the activity [38,39], and only one didn't use any evaluation methods [34]. Table 2 shows how almost all of the authoring tools, using various methods of evaluation, presented similar conclusions, which are often described in terms of being intuitive.

Table 2. Detecting “intuitiveness” in the virtual reality authoring tools.

Ref	Intuitiveness Quote
[32]	“[...] users can perform search more quickly and intuitively [...]”
[33]	“[...] rapidly create haptic feedback after a short training session.”
[34]	“The tool features an intuitive and easy to use graphical user interface appropriate for non-expert users”
[35]	“[...] positive feedback from users regarding ease of use and acceptability.” “[...] an authoring tool that is intuitive and easy to use.”
[36]	“[...] most participants commented positively on this application and [...] expressed that the application is easier for beginners”
[37]	“AffordIt! offers an intuitive solution [...], show high usability with low workload ratings”
[38]	“[...] people with little or even no experience [...] can install VAIF and build interaction scenes from scratch, with relatively low rates of encountering problem episodes”
[8]	“FlowMatic introduces [...] intuitive interactions, [...], reducing complexity, and programmatically creating/destroying objects in a scene.”
[17]	“[...] efficient data structure, for simple creation, easy maintenance and fast traversal [...] users can create VR training scenarios without advanced programming knowledge”
[39]	“[...] An immersive nugget-based approach is facilitating the authoring of VR learning content for laymen authors.”
[10]	“The interaction procedures are simple, easy to understand and use, and don't demand any specific skill expertise from users.”
[40]	“We choose user interface elements [...] to minimize learning time [...]” “it was useful to see each others' work in real time to improve workspace awareness, and it was easy to share findings with one another”
[41]	“Evaluation results indicate the positive adoption of non-experts in programming [...] participants felt somewhat comfortable using the system, considering it also as simple to use.”
[12]	“[...] we have analyzed the effectiveness, efficiency, and user satisfaction of VREUD which shows promising results to empower novices in creating their interactive VR scenes.”

3.3. Defining the Design Guidelines

The 14 studies were examined in relation to the research question “Q3: What are the guidelines for designing intuitive virtual reality authoring tools?” and the results are presented in the following section.

According to [22], getting the right VR specifications is difficult, and a creator's first few projects often fail. Well-designed VR may increase performance and save costs, give new worlds to explore, boost education, and develop deeper comprehension by letting users walk in someone else's shoes. However, even VR professionals can't always effectively define a new project from the start since good VR design combines technology and human perception. Trying to mitigate this challenge, [22] brings design guidelines to help authors create better VR experiences, which is the same strategy we use in this work but in a different context. As well as [21] we gathered design guidelines to help software developers, but specifically aiming to support the development of intuitive virtual reality authoring tools.

Thereafter, the design guidelines will help beginner authors to break the barrier of starting to create in virtual reality. Also, since most of the authoring tools found in the systematic review are just proof-of-concept, the guidelines can encourage the development of mainstream platforms with fewer limitations, democratizing the technology and increasing its maturity. Figure 2 shows how software developers can use design guidelines during the development process of authoring tools:

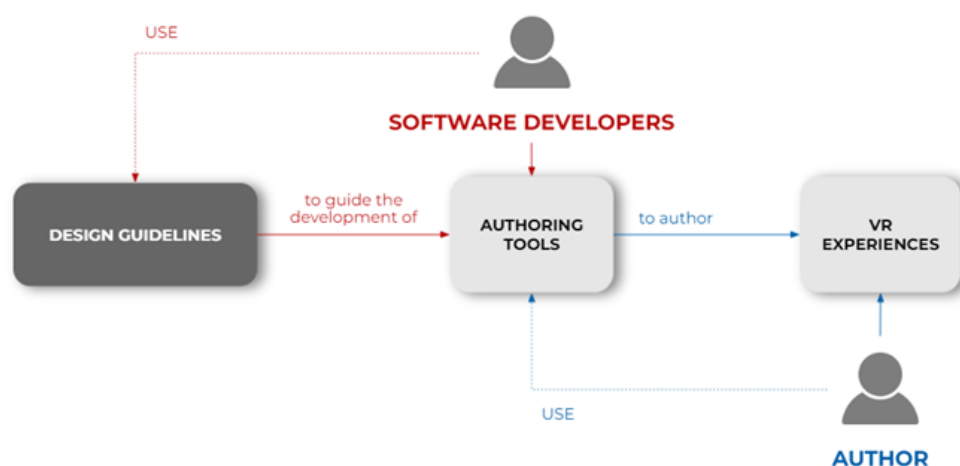


Figure 2. Information flow of the design guidelines application in the authoring tool life cycle .

Ref. [16] discusses the lack of ontologies related to the concepts of virtual reality authoring tools, indicating that there are few connected standards for the development of these platforms. In other words: concepts, methods, and nomenclature are not well-established, resulting in the development of authoring tools with vastly different formats and the application of diverse evaluation techniques to determine their usability. Similarly, [43] identified the need for a taxonomy proposal for the metaverse, since the wide scope of this concept causes a lack of understanding about how it works. Between the proposed taxonomies, we can point out the "components" that are thought to be necessary for the realization of the metaverse, they were: hardware, software, and contents. We found a lot of similarities between the design guidelines suggested in this work and the technologies that have recently become issues and interests to the metaverse and were mapped by [43] as hardware, software, and content. This adds to the belief that the guidelines can positively contribute to the creation of the metaverse, through their influence in facilitating the use of the components that form them.

Moreover, our findings also contribute to initiating or advancing the creation of ontologies for the development of virtual reality authoring tools in relation to the gap identified previously by [16]. Due to the lack of ontologies for authoring tools, the concepts and common functions among the authoring tools analyzed, often used different terms to refer to the same element. It is important to highlight that the guidelines obtained complement each other, and were never presented in isolation. The non-identification of a guideline in a given work does not mean that the authoring tool does not use it, only means that it was not mentioned in the article description. Also, the identification of a guideline in the article is not necessarily linked to its presence in the tool, but it may have been cited as

an application of previous work or an intention to improve the tool in the future. Table 3 summarizes the design guidelines and their application.

Table 3. Design Guidelines list, classification, articles and frequent terms

N	Design Guidelines	Classif.	Articles	Frequent Terms
1	Adaptation and commonality	Requirement	[8,12,17,34,35,38–41]	interoperability, exchange, data type, patterns, multiple, modular, export/import process, hardware compatibility
2	Automation		[10,12,17,32,37,38,41]	inputs, artificial intelligence, algorithms, translation, reconstruction, active learning, human-in-the-loop, neural systems
3	Customization		[8,10,12,34–41]	control, flexibility, interactions, manipulate, change, transformation, adapt, modify, programming, editing, modification
4	Democratization		[8,12,34,38–41]	web-based, popularization, open-source, free assets, A-FRAME, WebGL, deployment
5	Metaphors		[8,10,12,17,32–34,36,37,39,40]	natural, organic, real life, real-world, physicality, abstraction; embodied cognition
6	Movement Freedom		[8,10,32,34,36,37,39,40]	manipulation, gestures, position, unrestricted, selection, interaction, flexible, free-form
7	Optimization and Diversity Balance		[8,10,12,17,32,35,37,39–41]	trade-off, less steps, fast, complete, limitation, effective, efficient, simplify, focus, priorities
8	Documentation and Tutorials	Feature	[8,12,17,37,38,41]	help, support, fix, step-by-step, learning, practice, knowledge, instructions
9	Immersive Authoring		[8,10,12,17,32,34–37,39,40]	WYSIWYG, engagement, 3D modeling, programming, 3D interaction, paradigm, creation, HMD

Table 3. Cont.

N	Design Guidelines	Classif.	Articles	Frequent Terms
10	Immersive Feedback		[33,35–37]	visual, haptic, hardware, multisensory, physical stimuli, senses
11	Real-time Feedback		[8,12,17,32–37,39,40]	simultaneous, latency, WYSIWYG, synchronization, preview, immediate, run-mode, liveness, compilation, direct
12	Reutilization		[8,10,12,17,32,34,36,38,39,41]	retrieve, assets, objects, behaviors, reusable, patterns, store, library, collection, search
13	Sharing and Collaboration		[12,34,35,40,41]	multi-user, multi-player, remote interaction, community, simultaneous, communication, network, workspace
14	Visual Programming		[8,17,36,39,41]	primitives, logic, dataflow, nodes, blocks, modular, prototype, graphic

The next sections describe the design guidelines classified as requirements.

3.3.1. Adaptation and Commonality

This guideline relates to “interoperability” [43], enable the integration of different data acquisition sources (hardware or software), adaptable to a wide variety of cases and purposes; being usable for any application field (education, science, history, business, culture, design); allow communication with different types of VR hardware, like different HMD, controllers, and wearables like haptic gloves and clothes; use patterns, blocks, nodes, or modules to organize functionalities; allow content creators to set up different modules by turning on and off plugins, which can also change the tool’s user interface (UI); use the same tool to create for different platforms, such as personal computers, head-mounted displays, and mobile smartphones; use common programming languages and/or have the ability to use a known language of the user’s choice; accept different file extensions for the same type of data, such as .fbx, .catpart, and .igs, which are all extensions for 3D data from different types of 3D modeling; having a unique file extension that could retain all kinds of information and be used by all software would be the perfect scenario, which is done by Universal Scene Description (USD) files, an extension created by Pixar. The following examples illustrate the guideline:

- “To ensure a proper multisensory delivery, the authoring tool must communicate effectively with the output devices” [35];
- “using semantically data from heterogeneous resources” [17];
- “Establishing an exchange format and standardizing the concept of VR nuggets is a next step that can help to make it accessible for a greater community” [39].

3.3.2. Automation

This guideline concerns to automatic processing of activities that would require human interference, the algorithms must complement the human creative work and avoid non-productive

activities; use of an AI network like CNN or GAN to create systems that can analyze inputs and come up with better results; use of simple sketch drawings to search for equivalent 3D models; scan the physical world through complementary hardware like LiDARs or smartphones and use a raw point cloud to retrieve better virtual models; produce 3D models out of 2D images; provide autonomous tools to segment the 3D mesh into minor parts; triangle reduction on high polygon objects; follow human repetitive activities to create codes to reproduce them (human-in-the-loop); prediction of actions with smart suggestions, such as adding functions and behaviors to objects; AI assistant to provide tutorials and help as needed; use various inputs, such as voice commands, to activate a functionality [43]; translation into different languages; using cameras to track the authors' bodies so that the movements can be analyzed and behaviors can be made automatically. The following examples illustrate the guideline:

- "The number of triangles on high polygon objects were reduced to optimize the cutting time to an order of magnitude of seconds" [37];
- "In other words, the interaction manager enables developers to create events that are easy to configure and are applied automatically to the characters" [38];
- "The idea is to provide users with a modeling tool that intuitively uses the reality scene as a modeling reference for derivative scene reconstruction with added interactive functionalities" [10].

3.3.3. Customization

This guideline refers to give the author enough control over changes; apply 3D content anywhere in the virtual environment; align the 3D virtual models to scale, position and orientation; configure the appearance of 2D and 3D elements, changing color, size and shape; assign behaviors and animations to a 3D mesh; assign and combine functions and interactions between objects; modify scene lighting, cameras, and environments; set timing, duration, start/end point, intensity, and direction of VR multisensory stimuli; create more expressive interactivity through action specification for VR hardware, such as controllers and/or haptic gloves; apply emotional state and personality to a virtual agent; specify behaviors to react to discrete events such as user actions, system timers or collisions; customize the VR hardware while the software is always in an executable state; add annotations; edit texts; set pattern-specific parameters on software that uses them; organize the workspace layout by changing tools, tabs, and windows of the software. The following examples illustrate the guideline:

- "While the state-of-the-art immersive authoring tools allow users to define the behaviors of existing objects in the scene, they cannot dynamically operate on 3D objects, which means that users are not able to author scenes that can programmatically create or destroy objects, react to system events, or perform discrete actions" [8] - missing customization;
- "The system workflow design of VRFromX that enables creation of interactive VR scenes [...] establishing functionalities and logical connections among virtual contents" [10];
- "Some requests were [...] more freedom to change the parameters of the experience, i.e., to right click on 3D models and change the parameters of the assets on the fly" [41].

3.3.4. Democratization

This guideline relates to provide people with access to technical expertise via a radically simplified experience; authoring tools that can be accessed via a web browser (web-based) make it easier for people to get started because they don't require the download of a special program and can be used for a variety of cases and purposes; web-browser application can also be used through mobile devices that support VR, bringing more accessibility and knowledge about VR development; open-source and publicly available tools can reach multiple researchers to build and evaluate them; use of platforms such as GitHub to share resources and encourage users to contribute their own assets; empowerment of the citizen-developer model, with no-code procedures to design and develop VR applications;

hardware popularization with lower costs and better quality; use of free stores for the distribution of applications and plug-ins at no cost; use of libraries and frameworks such as Three.js and A-FRAME for web-browser development. The following examples illustrate the guideline:

- “[...] the advances of WebVR have also given rise to libraries and frameworks such as Three.js and A-FRAME, which enable developers to build VR scenes as web applications that can be loaded by web browsers” [8];
- “FlowMatic is open source and publicly available for other researchers to build on and evaluate” [8];
- “[...] democratization is focused on providing people with access to technical expertise (application development) via a radically simplified experience and without requiring extensive and costly training” [41].

3.3.5. Metaphors [21]

This guideline refers to turn abstract concepts into tangible tools; use of visual resources and gestures to execute actions in the virtual world in a similar way to the real world, which improves the author's immersion; begin actions with natural interactions and manipulation, for example, inserting a virtual disk into a virtual player as a start trigger to play music; move and position objects as if they were in the real world; use of buttons on the controllers to reproduce actions similar to what we would do in real life, like pulling the trigger button to grab an item and releasing it to drop; use of miniatures to localize things at a glance on the interface; connect objects distant from each other by making the physical movement of drawing visible lines between them; use of visual icons, such as fire and ice, to represent haptic feedbacks, like warm and cold, to the user, inducing a multisensorial experience; use different shapes and colors to represent different types of data; use of numbers to indicate sequences; use of hologram overlays to show the content of a pattern or object before interacting with it; real-world and virtual events can be linked using IoT-enabled devices, for example, starting an object print in a virtual printer can start the process on a physical 3D printer; not using the correct metaphor can sometimes lead to a user misinterpreting the tool or action; in different contexts like collaborative work, metaphors can naturally appear, such as the formation of individual territories when working in groups in the same space. The following examples illustrate the guideline:

- “They can draw edges to and from these abstract models to specify dependencies and behaviors (for example, to specify the dynamics of where it should appear in the scene when it shows up)” [8];
- “Similar to Alice in Wonderland, the users will gradually shrink as they trigger the entry procedure. Authors can access the world in miniature model and experience it in full scale to make changes to the content” [39];
- “Compared to the logic used in the construction of interactions, the task construction uses generic activities which should be also clear to novices without a technical background, since they are comparable to actions in the real world” [12].

3.3.6. Movement Freedom

This guideline concerns to the use of body movements to simplify creation and interaction when immersively authoring in virtual reality; use of 3D hand drawing (not only 2D) to retrieve 3D models, even with non-perfect sketches; freedom to explore the space, touch objects, manipulate elements, and encounter other users in a flexible virtual space; have the ability to move freely and safely in the virtual world, zoom in and out without needing to change positions in the physical world; immersively edit programming elements through direct manipulation; manipulate virtual objects using movements similar to those in the physical world, which can also be interpreted as a metaphor; freely arrange elements anywhere in the virtual space; interact with and edit 3D elements through simple hand gestures in a free-form manner, for example, by selecting an area of the 3D to be cut; create organic

3D shapes through immersive modeling; organize a workspace using all the extensions of a virtual environment; have the option to work individually but also access another user by moving toward them to share items or communicate; have different options to visualize all the extensions of an element, either by rotating it, physically moving around it, or even going through it to have different points of view. The following examples illustrate the guideline:

- “One reason is that through direct manipulation users can feel more immersed—as if the wire is in their hands” [36];
- “A brush tool was developed which enables users to select regions on point cloud or sketch in mid-air in a free-form manner” [10];
- “Users can also perform simple hand gestures to grab and alter the position, orientation and scale of the virtual models based on their requirements” [10].

3.3.7. Optimization and Diversity Balance

This guideline relates to the reduction of steps to authoring experiences without limiting creative freedom, which can often be achieved with the application of other guidelines such as automation, visual programming, and reutilization; give the authors the feeling of completing more activities in less time, by reducing the number of inputs to get a result; reduction of ambiguity between views in 2D and 3D by authoring in immersion, so the user doesn't have to spend a lot of time imagining projections; improve the efficiency of the editing process through collaborative work with many users; use of a programming language that is easy to use and has free codes available from outside libraries; position priority items physically close to the user, like keeping a set of tools always attached to the author's hand; avoid complexity and unnecessary actions, which can lead to incomprehension, impatience, and fatigue for authors; don't show all training materials at once to reduce cognitive load; organize functionalities in patterns and categories to focus attention during development; use the right rendering modes and make the best use of the hardware to always get good graphics and performance; combine simple elements to create others that are more complex. The following examples illustrate the guideline:

- “To make our system more efficient, we have to limit the capabilities of the Action entity targeting simple but commonly used tasks in training” [17];
- “The construction uses two dialogs to create the task and the activities so that the novice only needs to focus on the current task or activity” [12];
- “We decreased further the complexity by using wizards to focus the user on smaller steps in the development” [12].

The following sections describe the design guidelines classified as features.

3.3.8. Documentation and Tutorials [21]

This guideline refers to educate the user while using a tool, demonstrating the step-by-step process in real time; use diversified resources to present how to execute a function, such as images, animations, recorded videos, text, audio guidance, holographic icons, and virtual embodied characters; create specific initial tasks to teach basic tools on practice; publish tutorials in a variety of places, including YouTube, software documentation, and online forums; make sure to include missing information reported by users to complement the materials; encourage online communities to create more knowledge about the tool; include error messages to help the user understand what not to do and how to recover activities; make sure that help buttons are visible and easily accessible; avoid the presentation of too many steps at once, keeping enough details and a logical structure to follow; use automation to detect when the user is having difficulties to move on with a task and provide an insight to solve that. The following examples illustrate the guideline:

- “For each step, instructions are visualized as text in the menu to help participants remember which step they are performing” [37];

- “We believe that more visual aid in the form of animations showing the movement path can help ease the thinking process of participants” [37];
- “Documentation would be another interesting direction in the future, as two participants said they preferred A-FRAME in the sense that the APIs documentation was detailed and easy to understand” [8].

3.3.9. Immersive Authoring

This guideline regards to avoid 2D-display or projections while creating a virtual world; perform multiple activities while immersed and use the immersion to improve the author’s creation experience by, for example, executing a sketch in 3D to start a search for assets, 3D modeling, programming, building scenes or environments, reading documentation, and interacting with other authors; enjoying an immersive experience that has been deployed is not the same as creating this experience using virtual reality as a development tool, as the first option is only available to the final user; when applied with real-time feedback, immersive authoring creates a “what you see is what you get” (WYSWYG) experience; reduce the abstraction needed to convert 2D information to 3D; allow users to share information and resources while they are in the same space and working with other people; a good immersive authoring interaction in virtual reality is heavily influenced by movement freedom and haptic feedback; to fit with the user view extension and avoid visual pollution or confusion, the immersive user interface must be simplified; avoid switching back and forth between the 2D and 3D screens to check how things are displayed in immersion; well applied to testing VR functions in real-time and debugging; one issue is that wearing a HMD for an extended period of time can be exhausting. The following examples illustrate the guideline:

- “[...] expedites the process of creating immersive multisensory content, with real-time calibration of the stimuli, creating a “what you see is what you get (WYSWYG)” experience” [35];
- “[...] immersive authoring tools can leverage our natural spatial reasoning capabilities” [36];
- “With the lack of additional spatial information and the disconnection between developing environments (2D displays) and testing environments (3D worlds), users have to mentally translate between 3D objects and their 2D projections and predict how their code will execute in VR” [8] - (missing immersive authoring).

3.3.10. Immersive Feedback

This guideline relates that, in virtual reality, action feedback is both visual and haptic/physical, using hardware parts, such as HMD, controllers, and wearables as an extra interaction source; immersive experience feedback can have multiple formats, from rendered icons and symbols to haptic stimuli like controller vibrations; it’s possible to accurately represent physical stimuli such as thermal, vibrotactile, and airflow, which require different hardware to reproduce and can be costly or inflexible, but would increase immersion [43]; users accept creative solutions such as animations, sounds, and icons representing physical stimuli as feedback; visual and haptic feedback must occur in real-time to be accurately felt, or the user will not engage with the application if the stimuli arrive at the wrong time; the tool must allow the author to apply immersive feedback and preview the results before releasing them; associating behaviors from virtual elements to the tracking of hardware (HMD and controllers); configuring controller buttons to start logical operations to facilitate the development, such as activating a virtual menu attached to the hand or a frequently used function. The following examples illustrate the guideline:

- “Rendering haptic feedback in virtual reality is a common approach to enhancing the immersion of virtual reality content” [33];
- “[...] various types of haptic feedback, such as thermal, vibrotactile, and airflow, are included; each was presented with a 2D iconic pattern. According to the type of haptic feedback, different properties, such as the intensity and frequency of the vibrotactile feedback, and the direction of the airflow feedback, are considered” [33];

- “The use of multisensory support is justified by the fact that the more the senses engaged in a VR application, the better and more effective is the experience” [35].

3.3.11. Real-time Feedback [21]

This guideline relates to a real-time visualization or physical perception of what is being authored, related either to 3D editing, code compilation, animation preview, or hardware set-up used for a scene; help avoid making mistakes while creating, as you don't need to wait until the end to see the result; minimizing latency [43]; it allows non-experts to spot mistakes much more quickly; visual representation of actions performed on objects, such as a wireframe highlight to describe the geometry selection and an animation preview to show if the behaviors attached to an object really work; view the editing actions of other users in collaborative sessions as they occur simultaneously; preview of multisensory physical stimuli, such as wind, heat, or vibration, while applying them to objects, despite the fact that they are frequently created through code in a 2D screen; available either for conventional 2D monitors or HMD devices; allows better fine-tuning of the experience; when associated with immersive authoring, real-time feedback enables content creators to have a “what you see is what you get” experience, which means the user has a real view of the virtual environment while composing the scene; the authoring tool must allow the author to choose between turning on or off this feature as it can often cause issues and delay during the initialization of complex scenarios due to the quantity of information. The following examples illustrate the guideline:

- “AffordIt! offers an intuitive solution that allows a user to select a region of interest for the mesh cutter tool, assign an intrinsic behavior and view an animation preview of their work” [37];
- “We believe that more visual aid in the form of animations showing the movement path can help ease the thinking process of participants” [37];
- “The novices are supported in the construction by visualizing the interactive VR scene in the development. This ensures direct feedback of added entities to the scene and modified representative parameters of the entities inside the scene. This enables the novice to spot mistakes immediately” [12].

3.3.12. Reutilization

This guideline concerns to optimize development time by retrieving relevant elements from a collection or library, such as 2D/3D objects, audio files, codes to set behaviors and interactions, animations, lighting, etc., so the author doesn't always need to have advanced knowledge in 3D modeling or programming [43]; libraries and collections must be integrated into the software so the user doesn't need to access external sources and go through different processes to import different formats of files to the authoring tool, facilitating scene creation; integrating popular libraries into the tool leads to a bigger variety of models, considering that more authors are collaborating with these libraries; it's difficult to find the right element in large libraries; automation processes, such as AI networks trained to search for 3D assets in the virtual world using free-hand sketches, can help; saving author's creations for later is another form of reusing things; having templates helps start content creation in authoring tools; photogrammetry is an automated way to retrieve objects from the real world using cameras. The following examples illustrate the guideline:

- “We propose that by utilizing recent advances in virtual reality and by providing a guided experience, a user will more easily be able to retrieve relevant items from a collection of objects” [32];
- “[...] we propose intuitive interaction mechanisms for controlling programming primitives, abstracting and re-using behaviors” [8];
- “Users can also save the abstraction in the toolbox for future use by pressing a button on the controller” [8].

3.3.13. Sharing and Collaboration

his guideline relates to the creation and manipulation of virtual space via collaborative works in which multiple and disparate stakeholders can use their imaginations while working with multisensory immersion from a local or remote network; follow each other's activities in real-time; present ideas, products, and services to stakeholders, executives, or buyers in a business context; speed up the creation process with more workers dealing with different tasks at once; enable virtual round-tables for creative works, improving prototyping processes; combine the knowledge of different professionals in the same experience; edit of different objects at the same time by different users; people with more immersive technology experience can better assist and guide beginners while sharing the same space; users can change how others perceive them by customizing the color and shape of their avatars [43]; in sharing activities, tasks can be assigned and materials can be switched between users, like 3D and 2D assets, text documents, textures, etc; create specific tools to enable a better experience in collaborative mode, like setting mechanisms to lock the editing of an object by a user while it is being edited by another person; different groups of people will interact in different ways and at different levels and frequencies, changing the format of the discussions and also establishing social protocols such as owning objects and claiming territory in virtual space. The following examples illustrate the guideline:

- "[...] directly transmitted to others, and they can observe the doings of others in real-time. The users work together on a virtual scene where they can add, remove, and update 3D models" [34];
- "This is useful because multisensory VR experiences might require multiple features that are produced by different professionals, and a collaborative feature will enable to the entire team to work simultaneously" [35];
- "Each user is uniquely identified by a floating nameplate and avatar color. The same color is also used for shared brush selections. This allows users to see the actions of others to support collaborative tasks and information sharing, as well as to avoid physical collisions" [40].

3.3.14. Visual Programming

This guideline concerns to programming through dataflow instead of creating text lines of code to create behaviors and reactions for the scene components, characters, and objects, which leads to a reduction of text inputs; use geometrical formats as nodes that already have a function applied to them, so the author doesn't need to re-write the text or even know how to do it; all the functions and connections can be presented in a graphic and optimized way; the Blueprint from Unreal Engine, applied through 2D interaction, is a well-known and well-implemented visual programming format in the world of game engines; there are already a variety of formats for the Visual Programming Languages, and they can be implemented in both desktop editing mode (2D screen) and immersive authoring mode (HMD); the nodes containing pre-set functions can be called "primitives"; the connection between primitives is also visual, being usually represented by edges going from one node to the other; still, this format can have problems becoming too complex when the codes get too big; it helps with reutilization as the abstraction of functions as groups of nodes can be saved, duplicated, and united to create more complex functions; it can speed up the process of prototyping behaviors. The following examples illustrate the guideline:

- "FlowMatic uses novel visual representations to allow these primitives to be represented directly in VR" [8];
- "Unreal Blueprint, a mainstream platform for developing 3D applications, also uses event graphs and function calls to assist novices in programming interactive behaviors related to system events" [8];
- "The development of a visual scripting system as an assistive tool aimed to visualize the VR training scenario in a convenient way, if possible fit everything into one window. The simplicity of this tool was carefully measured to provide tools used also from non-programmers. From

the beginning of the project, one of the main design principles was to strategically abstract the software building blocks into basic elements" [17].

3.3.15. Additional considerations

Other factors were frequently mentioned in the reviewed works, such as engagement, fun-to-use, immersiveness, physical comfort, graphical quality, and acquisition cost of equipment. These factors are not directly involved with intuitiveness, as the guidelines are, because they could be considered as consequences or even challenges related to the democratization of virtual reality technology. Therefore, there is a distance from the design guidelines proposal for the development of more intuitive virtual reality authoring tools, but they are related to the general structure of the technology in our society, aiming for a more popular use among people with various levels of knowledge.

Concerning the term's definition, "engagement" stands for the active participation and involvement of the user with the tool, indicating that the users have a high level of engagement with the experience provided. As a complement to "engagement", the term "fun-to-use" is used to describe the tools with which the users were not only focused on completing the tasks but also having a good time doing it, it's frequently related to the concept of gamification. "Immersiveness" is the quality of an experience that provides the user with deep absorption and makes them feel like they've been transported to another world through multisensory feedback, not just based on images and sounds.

The lack of "physical comfort" is frequently mentioned since virtual reality equipment is heavy, so spending a lot of time with it can be exhausting and cause motion sickness in some people. "Graphical quality" is another factor pointed out as being missing from virtual reality experiences due to the geometry optimization necessary to be processed by the HMD. "Graphical quality" has a direct effect on how immersed you feel in an experience, since the better the graphics, the more immersed you feel. Finally, the high "cost" of good virtual reality equipment acquisition is pointed out as one of the main reasons why the technology has not been well popularized so far. The following are quotations from the reviewed works that used the terms:

- Engagement: "The system also provides an engaging and immersive experience inside VR through spatial and embodied interactions" [10];
- Fun-to-use: "[...] the results of the statement if the participant had fun constructing the interactive VR scene suggests that VREUD supplies novices with a playful construction of interactive VR scenes, which could motivate them to develop their first interactive VR scene" [12];
- Immersiveness: "[...] the majority of VR applications rely essentially on audio and video stimuli supported by desktop-like setups that do not allow to fully exploit all known VR benefits" [35];
- Physical comfort: "[...] some participants commented that navigating the virtual world could cause slight motion sickness" [8] / "[...] we could observe impatience and fatigue when the participants had to type in the text for the callouts using the immersive technology (a virtual keyboard) or had to connect the nuggets to bring them in chronological order" [39];
- Graphical quality: "One disadvantage of these tools is that they do not support highly photorealistic graphics and first person view edits which are achievable only by Unreal Engine and professional CAD software in runtime environments" [41];
- Cost: "It lowers the cost as the templates and the abstractions replace the Application Designer and Programmer by standardizing" [41] / "These wider application areas of VR require, besides affordable devices, a usable process of authoring to reach the full potential" [12].

Another key consideration is how, in practice, the design guidelines should positively contribute to the growth of the metaverse through their impact on the development of easier-to-use authoring tools and, consequently, the increase in the volume of virtual world creation.

Ref. [44] discusses the computing power and programming required to create virtual worlds and the accurate physical behavior of related objects. Ref. [23] presents in their metaverse Architecture the concept of metaverse Engines, which include software technologies used in the creation of virtual

worlds, such as immersive technologies (VR, AR, MR), Brain-Computer Interaction (BCI), Artificial Intelligence (AI), Digital Twins (3D creation), and Blockchain. Ideally, the metaverse engine would use big data coming from the real world in an automatic way to create, maintain, and update the virtual world. The virtual economy would come from virtual avatars doing things on their own, like trading personalized content made by AI to improve the metaverse ecology.

In contrast to [23] ideas, human developers are still in charge of making virtual worlds for the metaverse. Because of that, according to [44], "Virtual World Engines" will become a standard feature of the metaverse, much like English is a standard language in the world, as the global economy continues to shift to virtual worlds. And, beside the many advantages presented by mainstream game engines such as Unreal and Unity, there is still a lot of discussion on what is the easiest and best way to build the metaverse, facilitating exchanges of information, virtual goods, and currencies between these virtual worlds.

The Integrated Virtual World Platforms (IVWPs) are a new approach to dealing with the creation of virtual worlds that, according to [44], "are designed so that no actual "coding" is required. Instead, games, experiences, and virtual worlds are built using graphical interfaces, symbols, and objectives [...] The IVWP interface enables users to create more easily and with fewer people, less investment, and less expertise and skill."

This definition is very similar to those used to refer to authoring tools in several of the works analyzed here. But we can tell there is a difference by looking at the context in which each idea is used. While [44] delves into game development, bringing examples such as Roblox, Minecraft and Fortnite Creative, platforms that reach thousands of users and make thousands of dollars, authoring tools developed in the academic context are seen as proof-of-concepts with non-profit goals and are most often applied in the professional environment, not entertainment. Furthermore, it is interesting to see how both IVWPs and Authoring Tools share not only concepts but also challenges, such as the fact that both "wants to enable creators' creative flexibility while standardizing underlying technologies, maximizing interconnectivity among everything that's built, and minimizing the need for training or programming knowledge on the part of creators" [44]. Therefore, these platforms are more difficult to develop than the Game Engines mentioned above, as every feature becomes a priority.

Facilitating virtual reality development is also not a priority in the mainstream when it comes to IVWP, since one of the biggest platforms in the middle, focused on VR and AR, Facebook's Horizon World, remains small when compared to Roblox, which provides immersive VR but prioritizes traditional screens [43].

As for which platform to use for the metaverse, [44] concludes that, due to the diversity of potential applications, the high technical level of difficulty to unite all of them in something unique, and given the speed at which new platforms are emerging, the best solution will be to handle all existing tooling options simultaneously, also avoiding market monopolization by a single corporation. That's why it's possible to say that gathering design guidelines could also affect the development of metaverses, since it should help make authoring tools, or even IVWP, that are more intuitive for the people who make virtual worlds to use.

4. Conclusions

The 14 gathered design guidelines can support the development of more intuitive authoring tools focused on authors with limited prior knowledge of 3D modeling and programming, resulting in a more sustainable virtual reality. With the evolution of immersive technologies, many of these guidelines are becoming easier to implement. However, it's also important to understand the intended audience and demand so the priority guidelines for that context can be defined. Even so, the main contribution of this research is the systematic organization and classification of widely used themes and concepts in virtual reality, since none of them were invented in this research, creating, in other words, ontologies.

As a limitation of this study, the design guidelines were derived from the reviewed articles, which means other guidelines may not be identified by our literature search. In addition, many of the guidelines are also connected to software development principles in general, therefore, some of them could be applied to applications not related to virtual reality. Also, our categorization of the guidelines is subjective; they could be organized in different categories.

Concerning future research, each guideline provides the opportunity to delve deeper into the definition in a technical software development approach, even to the point of creating subclasses. Since the guidelines are never presented in isolation, it would be interesting to analyze the correlation level between them, to better understand how to apply them in a project. In addition, it is necessary to conduct actual tests in the context of application development using the guidelines in order to comprehend their impact on the project definition. Finally, it is possible to test the use of the guidelines as a system for evaluating existing authoring tools from the perspective of novice professionals or individuals with no background in 3D modeling or programming.

Above all, this research focused on the democratization of tools for creating virtual worlds, which directly impacts the faster and sustainable advance and dissemination of trends like the metaverse. Because of this, virtual reality technology will keep helping to reach the UN Sustainable Development Goals by giving more people the chance and independence to create immersive experiences and develop skills for the digital transformations of society.

Supplementary Materials: The following supporting information can be downloaded at the website of this paper posted on Preprints.org.

Author Contributions: Conceptualization, I.L.C. and I.W.; methodology, I.L.C. and I.W.; validation, I.W., A.L.A.J., T.B.M. and C.V.F.; formal analysis, I.W. and A.L.A.J.; investigation, I.L.C.; resources, I.W., A.L.A.J. and T.B.M.; data curation, I.L.C.; writing—original draft preparation, I.L.C.; writing—review and editing, I.W., A.L.A.J., T.B.M. and C.V.F.; visualization, I.L.C.; supervision, I.W., A.L.A.J. and T.B.M.; project administration, I.L.C. and I.W. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The data presented in this study is fully available in the Supplementary Material.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Matt, C.; Hess, T.; Benlian, A. Digital transformation strategies. *Business & information systems engineering* **2015**, *57*, 339–343.
2. Ebert, C.; Duarte, C.H.C. Digital transformation. *IEEE Softw.* **2018**, *35*, 16–21.
3. Scurati, G.W.; Bertoni, M.; Graziosi, S.; Ferrise, F. Exploring the use of virtual reality to support environmentally sustainable behavior: A framework to design experiences. *Sustainability* **2021**, *13*, 943.
4. de Freitas, F.V.; Gomes, M.V.M.; Winkler, I. Benefits and challenges of virtual-reality-based industrial usability testing and design reviews: a patents landscape and literature review. *Applied Sciences* **2022**, *12*, 1755.
5. da Silva, A.G.; Mendes Gomes, M.V.; Winkler, I. Virtual Reality and Digital Human Modeling for Ergonomic Assessment in Industrial Product Development: A Patent and Literature Review. *Applied Sciences* **2022**, *12*, 1084.
6. Cassola, F.; Pinto, M.; Mendes, D.; Morgado, L.; Coelho, A.; Paredes, H. A novel tool for immersive authoring of experiential learning in virtual reality. 2021 IEEE Conference on Virtual Reality and 3D User Interfaces Abstracts and Workshops (VRW). IEEE, 2021, pp. 44–49.
7. Nebeling, M.; Speicher, M. The trouble with augmented reality/virtual reality authoring tools. 2018 IEEE international symposium on mixed and augmented reality adjunct (ISMAR-Adjunct). IEEE, 2018, pp. 333–337.

8. Zhang, L.; Oney, S. Flowmatic: An immersive authoring tool for creating interactive scenes in virtual reality. *Proceedings of the 33rd Annual ACM Symposium on User Interface Software and Technology*, 2020, pp. 342–353.
9. Sherman, W.R.; Craig, A.B. *Understanding virtual reality: Interface, application, and design*; Morgan Kaufmann, 2018.
10. Ipsita, A.; Li, H.; Duan, R.; Cao, Y.; Chidambaram, S.; Liu, M.; Ramani, K. VRFromX: from scanned reality to interactive virtual experience with human-in-the-loop. *Extended Abstracts of the 2021 CHI Conference on Human Factors in Computing Systems*, 2021, pp. 1–7.
11. Krauß, V.; Boden, A.; Oppermann, L.; Reiners, R. Current practices, challenges, and design implications for collaborative AR/VR application development. *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems*, 2021, pp. 1–15.
12. Yigitbas, E.; Klauke, J.; Gottschalk, S.; Engels, G. VREUD-an end-user development tool to simplify the creation of interactive VR scenes. *2021 IEEE Symposium on Visual Languages and Human-Centric Computing (VL/HCC)*. IEEE, 2021, pp. 1–10.
13. Berns, A.; Reyes Sánchez, S.; Ruiz Rube, I. Virtual reality authoring tools for teachers to create novel and immersive learning scenarios. *Eighth International Conference on Technological Ecosystems for Enhancing Multiculturality*, 2020, pp. 896–900.
14. Velez, D.; Zlateva, P. Virtual reality challenges in education and training. *International Journal of Learning and Teaching* **2017**, 3, 33–37.
15. Vergara, D.; Extremera, J.; Rubio, M.P.; Dávila, L.P. Meaningful learning through virtual reality learning environments: A case study in materials engineering. *Applied Sciences* **2019**, 9, 4625.
16. Coelho, H.; Monteiro, P.; Gonçalves, G.; Melo, M.; Bessa, M. Authoring tools for virtual reality experiences: a systematic review. *Multimedia Tools and Applications* **2022**, pp. 1–24.
17. Zikas, P.; Papagiannakis, G.; Lydatakis, N.; Kateros, S.; Ntoa, S.; Adami, I.; Stephanidis, C. Immersive visual scripting based on VR software design patterns for experiential training. *The Visual Computer* **2020**, 36, 1965–1977.
18. Lynch, T.; Ghergulescu, I. Review of virtual labs as the emerging technologies for teaching STEM subjects. *INTED2017 Proc. 11th Int. Technol. Educ. Dev. Conf*, 2017, Vol. 6, pp. 6082–6091.
19. O'Connor, E.A.; Domingo, J. A practical guide, with theoretical underpinnings, for creating effective virtual reality learning environments. *Journal of Educational Technology Systems* **2017**, 45, 343–364.
20. Ashtari, N.; Bunt, A.; McGrenere, J.; Nebeling, M.; Chilana, P.K. Creating augmented and virtual reality applications: Current practices, challenges, and opportunities. *Proceedings of the 2020 CHI conference on human factors in computing systems*, 2020, pp. 1–13.
21. Pressman, R.S. *Software engineering: a practitioner's approach*; Palgrave macmillan, 2021.
22. Jerald, J. *The VR book: Human-centered design for virtual reality*; Morgan & Claypool, 2015.
23. Wang, Y.; Su, Z.; Zhang, N.; Xing, R.; Liu, D.; Luan, T.H.; Shen, X. A survey on metaverse: Fundamentals, security, and privacy. *IEEE Communications Surveys & Tutorials* **2022**.
24. Page, M.J.; McKenzie, J.E.; Bossuyt, P.M.; Boutron, I.; Hoffmann, T.C.; Mulrow, C.D.; Shamseer, L.; Tetzlaff, J.M.; Akl, E.A.; Brennan, S.E.; others. The PRISMA 2020 statement: an updated guideline for reporting systematic reviews. *Systematic reviews* **2021**, 10, 1–11.
25. Booth, A.; Sutton, A.; Clowes, M.; Martyn-St James, M. *Systematic approaches to a successful literature review* **2021**.
26. How Scopus Works. Available online: <https://www.elsevier.com/solutions/scopus/how-scopus-works/content>. (Accessed on 7 December 2022).
27. Matthews, T. LibGuides: Resources for Librarians: Web of Science Coverage Details. Available online: <https://clarivate.libguides.com/librarianresources/coverage>. (Accessed on 7 December 2022).
28. Major HTC Vive VR “Breakthrough” to Be Shown at CES 2016. Available online: <https://www.techradar.com/news/wearables/major-htc-vive-breakthrough-to-be-unveiled-at-ces-2016-1311518>. (Accessed on 7 December 2022).
29. HTC Vive to Demo a “Very Big” Breakthrough in VR at CES. Available online: <https://www.engadget.com/2015-12-18-htc-vive-vr-big-breakthrough-ces.html>. (Accessed on 7 December 2022).
30. Brief history of Virtual Reality until the year 2020. Available online: <https://ez-360.com/brief-history-of-virtual-reality-until-the-year-2020/>. (Accessed on 7 December 2022).

31. Knapp, J.; Zeratsky, J.; Kowitz, B. *Sprint: How to solve big problems and test new ideas in just five days*; Simon and Schuster, 2016.
32. Giunchi, D.; James, S.; Steed, A. 3D sketching for interactive model retrieval in virtual reality. *Proceedings of the Joint Symposium on Computational Aesthetics and Sketch-Based Interfaces and Modeling and Non-Photorealistic Animation and Rendering*, 2018, pp. 1–12.
33. Chan, L.; Chen, M.H.; Ho, W.C.; Peiris, R.L.; Chen, B.Y. SeeingHaptics: Visualizations for communicating haptic designs. *Proceedings of the 21st International Conference on Human-Computer Interaction with Mobile Devices and Services*, 2019, pp. 1–5.
34. Capece, N.; Erra, U.; Losasso, G.; D’Andria, F. Design and implementation of a web-based collaborative authoring tool for the virtual reality. *2019 15th International Conference on Signal-Image Technology & Internet-Based Systems (SITIS)*. IEEE, 2019, pp. 603–610.
35. Coelho, H.; Melo, M.; Martins, J.; Bessa, M. Collaborative immersive authoring tool for real-time creation of multisensory VR experiences. *Multimedia Tools and Applications* **2019**, *78*, 19473–19493.
36. Zhang, L.; Oney, S. Studying the Benefits and Challenges of Immersive Dataflow Programming. *2019 IEEE Symposium on Visual Languages and Human-Centric Computing (VL/HCC)*. IEEE, 2019, pp. 223–227.
37. Masnadi, S.; González, A.N.V.; Williamson, B.; LaViola, J.J. Affordit!: A tool for authoring object component behavior in vr. *2020 IEEE Conference on Virtual Reality and 3D User Interfaces Abstracts and Workshops (VRW)*. IEEE, 2020, pp. 740–741.
38. Novick, D.; Afravi, M.; Martinez, O.; Rodriguez, A.; Hinojos, L.J. Usability of the Virtual Agent Interaction Framework. *International Conference on Human-Computer Interaction*. Springer, 2020, pp. 123–134.
39. Horst, R.; Naraghi-Taghi-Off, R.; Rau, L.; Dörner, R. Bite-Sized Virtual Reality Learning Applications: A Pattern-Based Immersive Authoring Environment. *J. Univers. Comput. Sci.* **2020**, *26*, 947–971.
40. Lee, B.; Hu, X.; Cordeil, M.; Prouzeau, A.; Jenny, B.; Dwyer, T. Shared surfaces and spaces: Collaborative data visualisation in a co-located immersive environment. *IEEE Transactions on Visualization and Computer Graphics* **2020**, *27*, 1171–1181.
41. Ververidis, D.; Migkotzidis, P.; Nikolaidis, E.; Anastasovitis, E.; Papazoglou Chalikias, A.; Nikolopoulos, S.; Kompatsiaris, I. An authoring tool for democratizing the creation of high-quality VR experiences. *Virtual Reality* **2022**, *26*, 105–124.
42. Brooke, J. SUS: a retrospective. *Journal of usability studies* **2013**, *8*, 29–40.
43. Park, S.M.; Kim, Y.G. A Metaverse: Taxonomy, components, applications, and open challenges. *Ieee Access* **2022**, *10*, 4209–4251.
44. Ball, M. *The metaverse: and how it will revolutionize everything*; Liveright Publishing, 2022.

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.