Ananya Ipsita[1]
Smart Manufacturing, Division of Engineering and Technology, CIT, University of Michigan-Flint, Flint, MI, United States
email: aipsita@umich.edu

Ramesh Kaki
Department of Mechanical Engineering, Birla Institute of Technology and Science, Hyderabad, Telangana, India

Mayank Patel
School of Mechanical Engineering, Purdue University, West Lafayette, IN, United States

Asim Unmesh
School of Mechanical Engineering, Purdue University, West Lafayette, IN, United States

Kylie A. Peppler
School of Education, Donald Bren School of Information and Computer Sciences, University of California, Irvine, Irvine, CA, United States

Karthik Ramani
School of Mechanical Engineering, Elmore Family School of Electrical and Computer Engineering, Purdue University, West Lafayette, IN, United States

Interactive authoring of outcome-oriented lesson plans for immersive Virtual Reality training

Immersive Virtual Reality (iVR) applications have shown immense potential for skill training and learning in manufacturing. However, authoring of such applications requires technical expertise, which makes it difficult for educators to author instructions targeted at desired learning outcomes. We present FlowTrainer, an LLM-assisted interactive system to allow educators to author lesson plans for their iVR instruction based on desired goals. The authoring workflow is supported by Backward design to align the planned lesson based on the desired outcomes. We implemented a welding use case and conducted a user study with welding experts to test the effectiveness of the system in authoring outcome-oriented lesson plans. The study results showed that the system allowed users to plan lesson plans based on desired outcomes while reducing the time and technical expertise required for the authoring process. We believe that such efforts can allow widespread adoption of iVR solutions in manufacturing training to meet the workforce demands in the industry.

Keywords: Virtual Reality, Virtual Reality Welding Simulators, Backward design, Virtual Reality Training, Large Language Models, LLM, Lesson Planning

1 Introduction

The U.S. manufacturing sector is facing a shortage of skilled workforce due to demographic shifts with an aging and retiring workforce and global shifts due to increased supply chain resilience [1–3]. These shifts call for the need to prepare students for a globally competitive environment. To meet these demands, training efforts must equip future professionals with both conventional and digital manufacturing skills [2,4]. Immersive Virtual Reality (iVR) technology presents promising potential to enhance such efforts by providing immersive and interactive learning experiences without geographic constraints and at reduced cost, instructor time, and effort [5]. Despite these benefits, the adoption of VR in manufacturing education is limited, which is partly due to the challenges faced by subject matter experts and educators to create content for training purposes [6–10]. This research provides an interactive system to reduce the barriers to authoring iVR content with an aim to facilitate iVR adoption for widespread skill training and learning.

Owing to the heavy dependence of technical modeling and programming expertise required for iVR content creation [11,12], manufacturing educators typically collaborate with VR developers to author relevant content for their courses [11,13,14]. This collaboration, which typically happens on a contractual basis over a stipulated period, results in a packaged product at the end of the development cycle, with minimal room for variability in the final instruction design. Any demand for design modifications typically requires version change for the entire application, thereby limiting the scalability of instructions based on training requirements [15]. Moreover, this causes the subject matter experts (SMEs) to compromise with and plan the training programs around the rigid situation. To ensure that the iVR-based learning content is effective for achieving the desired goals, it is important that the content is tailored to the specific needs of different user groups and learning environments. Therefore, there is a need to propose workflows that empower SMEs with control over iVR instruction planning, catering to the varying needs of learners and the learning process.

Prior studies have provided workflows to author iVR content for skill training and learning. However, these works have either lacked focus on the reuse of content for varied requirements [16–21] or highlighted the reuse aspect of authoring at an interaction level rather than learning level [22,23]. The latter, while still useful, faces limitation to be used in educational context as they focus more on the technical aspects of the system design [13]. In contrast, our research provides an interactive workflow to authoring lesson plans for iVR-based instruction while still grounded in the pedagogical theories. In particular, the workflow utilizes the principles of Backward design which first identifies the desired learning outcomes and objectives, and then aligns the skills, assessment criteria and learning activities based on the intended learning outcomes [24,25].

We present ***FlowTrainer***, an interactive workflow to enable authoring of lesson plans targeted towards desired learning outcomes. For example, in a welding training context, a desired learning outcome may be that learners are able to perform Tee-joint welding

[1]Corresponding Author.
Version 1.18, August 10, 2026

 Copyright © 2026 by ASME

safely and accurately.

Using our workflow, SMEs can first author lesson plans using a web-based editor, and subsequently test the lesson plan in VR and iterate till the target lesson plan is achieved. The web-based editor provides a LLM (Large Language Model)-supported user interface where the user inputs can be guided interactively to generate a lesson plan by following a Backwards design approach. The process starts with defining the learning outcomes and then defining measurable objectives, skills and assessment criteria, and finally identifying the lesson plan in the form of a sequence of learning activities. In our work, the learning activities that act as the building blocks for the lesson plan are developed by the VR developers and stored in a library for educators to use based on the course requirements. These learning activities are categorized under the four instructional phases of *Introduction*, *Presentation*, *Practice*, and *Application*. These instruction phases provide a concrete framework for planning learning units [26–29] and, when combined together, present a scalable framework which can be utilized to design a single lesson or a series of lessons, based on the target requirements of the training process.

Using a welding use-case and based on our prior learning rationale extracted using Backward design [30], a comparative user study was conducted with 8 experienced welders to test the usability and effectiveness of the system components to design the instructional flow for iVR-based learning units as compared to a baseline condition. The baseline condition is the preceding version of the current system which utilizes a similar workflow, but does not provide interaction capabilities to the user [30]. From the user study, we aimed to answer the following research question.

- To what extent does the system support SMEs in authoring and adapting training scenarios based on varied learning objectives as compared to the baseline condition?

The user study results indicated that the current system enables SMEs to flexibly author training scenarios with ease and low mental workload under increasing complexity. Comparative analysis from survey results shows higher usability, lower cognitive workload and reduced time of authoring in using the current system as compared to the baseline condition. User feedback supported these findings by emphasizing the intuitive design, automatic task generation, and reduced effort in lesson creation. These characteristics make the current system suited for designing iVR training scenarios that must adapt to varied learning objectives, and thus positioning it as a scalable and cognitively accessible authoring tool for SMEs.

Thus, the contribution of our work are as follows:

- the system workflow design that enables SMEs to facilitate pedagogically driven planning of iVR-based learning units the Backward design approach [24,25], and
- the demonstration of the workflow design using a welding use case along with the evaluation and results from a comparative user study conducted with 8 experienced welders as compared to a baseline condition.

The article's structure is outlined as follows: In Section 2, relevant literature is presented. Section 3 introduces key terminology to facilitate comprehension of the work. The design rationale is then detailed in Section 4. Section 5 provides specific insights into the implementation details. The user study procedure is presented in Section 5 with the results described in subsection 5.3. In Section 6, the study results are discussed in greater detail. Finally, Section 7 presents the conclusion of the work.

2 Related Work

2.1 iVR authoring in Education. Authoring tools can address some challenges in developing iVR training applications by making content creation faster and more efficient [12,31]. However, designing training scenarios remains difficult due to low code reuse and the need for advanced programming skills. Content reuse can reduce effort by avoiding redundant asset development [22,23]. Only a few tools support asset reuse/import, enabling designers to build customized experiences, but most rely on high-level componentization that emphasizes technical complexity [13]. This often requires engineers with technical expertise and thus limits SMEs' ability to prepare VR training. Machine learning-based approaches have also been explored, such as Kumar et al.'s curriculum generation for medical students using deep learning and hybrid recommendation systems [32]. While accurate, these automated methods exclude SME input, reducing their applicability to training contexts.

User-friendly content authoring tools are crucial for SMEs to design VR training scenarios that leverage their domain knowledge. Such tools reduce development time and effort, thereby promoting wider adoption of VR in training [13]. Several approaches highlight this potential. Zikas et al. demonstrated how innovative platforms enable users with limited programming expertise to design virtual environments [33]. Cassola et al. developed methods for creating semantic VR scenarios without requiring advanced programming or 3D modeling skills [14]. These approaches empower trainers and trainees to build diverse immersive learning scenarios, including certification contexts. Other solutions emphasize usability through intuitive interfaces and immersive authoring. Coelho et al. proposed collaborative authoring tools that provide real-time feedback, reducing iterations and accelerating development [34,35]. Wolfartsberger and Niedermayr introduced an immersive platform allowing trainers to specify elements, facilitate practice, and assess trainee performance [36]. Similarly, VR4Health enables teachers to monitor learning progress, collect motivational data, and provide targeted feedback [37]. Cassola et al. also introduced a framework for SMEs to author procedural sequences, expanding the availability of reusable training components [38]. Lécuyer et al. simplified scenario creation by recording expert actions in VR and transforming them into reusable sequences [39]. In terms of 3D content modeling, Walczak et al. applied domain knowledge representation and semantic web techniques to standardize content meaning [13,40]. However, most of these tools were designed as stand-alone applications for specific domains, often neglecting pedagogical integration into the development cycle of iVR learning applications. This limitation has restricted their broader adoption in educational and training settings.

Unlike iVR applications in gaming and entertainment, effective use in professional training requires integrating expertise from multiple domains to design pedagogically sound scenarios [41,42]. Mahdi et al. propose a stepwise process involving teachers articulating needs, adapting environments, operationalizing scenarios, and conducting simulations. This is supported by a custom-built tool whose limited features restrict adoption [42]. Saunier et al. advocate role separation in scenario creation, distinguishing designers, job experts, educational specialists, and teachers [43]. Pedagogical perspectives further reinforce the potential of VR in training. Kavanagh et al. emphasize constructivist pedagogy and gamification in VR learning design [44]. Keskitalo highlights the value of simulation in education and training [45]. Kemanji et al. argue for dedicated pedagogical models tailored to VR [15]. Building on these insights, our method introduces a workflow design guided by pedagogical models, employing user-friendly components that allow SMEs to plan instructional content across varied training needs and address challenges in adopting VR technology in education.

2.2 LLM-based lesson plan authoring in Education. LLMs are increasingly used to support lesson planning and instructional design. *LessonPlanner* helps novice teachers create pedagogy-aligned plans using Gagné's Nine Events, improving quality and reducing workload [46]. Hu et al. guide LLMs to simulate teacher-student interactions and generate reflective feedback to iteratively enhance high school math plans [47]. GPT-4 has demonstrated strong capabilities in setting objectives and organizing instructional content, though limitations remain in geometry and interdisciplinary modules [48]. Zheng et al. introduced a three-stage process using Retrieval-Augmented Generation and self-critique prompting to generate high-quality lesson plans across elementary math topics [49]. GenAI tools with interactive mega-prompts enable tailored lesson plans based on classroom demographics and learning goals, leading to improved inclusivity and time savings [50]. Additionally, agent-based approaches have improved alignment between generated content and instructional frameworks like Gagné's [51]. Our work builds on these approaches by introducing *FlowTrainer*, which extends LLM-based lesson planning to immersive VR-based manufacturing training, supporting outcome-aligned instruction in technical domains.

3 Clarification of Terms

The following definitions will aid in the understanding of terminology used in this research work.

- *Learning outcomes* indicate the relevant objectives (e.g., established goals, desired understandings) that will be addressed by the system. These will be useful in determining what students will know (e.g., what critical knowledge and skills will be acquired) and what students will be able to do as a result of the knowledge and skill acquisition [52]. For example, a learning outcome in the welding use case may be that learners are able to safely perform Tee-joint welding using a MIG welding process.
- *Learning objectives* are goals that target specific knowledge, skills, and dispositions taught in specific sections of a module [53]. For example, a learning objective may be to identify the components of a MIG welding machine or demonstrate the correct setup procedure for a Tee-joint welding task.
- *Learning plan* is a description of how to design the learning experience and instructions to achieve the desired outcomes in learning [53]. For example, a learning plan may include an introduction to welding safety, a demonstration of welding equipment, guided welding practice, and a final welding assessment task.
- *Learning activity* can be defined as a specific interaction of learner(s) using specific tools and resources, orientated towards specific outcomes [54]. For example, a learning activity may involve watching an avatar demonstration of Tee-joint welding, identifying personal protective equipment, or performing a virtual welding exercise.
- *Learning module or unit* is an organized collection of teaching materials consisting of behavioral objectives, a sequence of learning activities, and provisions for evaluation [55]. For example, a welding training module may include learning objectives, instructional videos, hands-on virtual welding activities, and performance assessments related to MIG welding.

4 Design Rationale

The process of instruction planning typically involves breaking down learning units or modules into segments of learning activities that revolve around relevant theories, concepts, or skills. Our proposed system for lesson planning of iVR-based instruction is based on the Backward design principles to effectively design outcome-oriented content [24,25]. According to the Backward design model, the process begins with identifying the learning outcomes, and then subsequently working "backwards" to identify the learning objectives to achieve the desired goals. The next step is to identify the skills required to achieve the outcomes and objectives, and appropriate assessment strategies to measure the skills during the various learning stages. The final step is designing the learning activities required to meet the goal. Our system follows a similar approach to guide instructors to interactively provide these parameters to come up with the required lesson plan which is the sequence of learning activities.

The planning of a learning unit is further guided by the standardized instructional framework that groups learning activities into the four instructional phases of *Introduction*, *Presentation*, *Practice*, and *Application*. This instructional framework is common across various instructional design models and outlines the entire process of designing a learning experience, from audience analysis to implementation and revision [26–29]. Furthermore, this instruction framework revolves around the learning outcomes, and prioritizes learner assessment and feedback throughout the process. Therefore, this can be easily integrated with the Backwards design framing of the curriculum [26]. We now briefly explain the four instructional phases utilized in the instruction planning process while providing examples of learning activities associated with each [56].

- ***Introduction:*** This phase serves as the introduction to the topic. Some guiding principles for designing learning activities are: (1) enable students to remember prior experiences they can build on, (2) set expectations by sharing learning outcomes, and (3) provide necessary hook to arouse emotions and feelings in learners to facilitate better learning and retention. Examples include stories, videos, readings, diagnostic quizzes, etc.
- ***Presentation:*** This phase guides students to engage with the content needed to achieve the learning objective. Some guiding principles for designing learning activities are: (1) determine what content is need-to-know vs. nice-to-know for achieving the objectives, (2) involve learners in actively learning the content within the situated learning context, and (3) make the content accessible and immersive for the learners. Examples include guided video tutorials, avatar demonstrations, narrations, and readings.
- ***Practice:*** This phase enables students to perform practice activities targeting understanding of the specific content and performance of skills. Some guiding principles for designing learning activities are: (1) model and scaffold skills using necessary cues and prompts, (2) provide support mechanisms in terms of cues, hints and prompts, and (3) associate built-in evaluation and feedback. Examples include quizzes, Q&A practice, and hands-on practical assignments.
- ***Application:*** This phase allows students to apply new content and skills to authentic situations and integrating this with previous knowledge. Some guiding principles for designing learning activities are: (1) mimic real-world tasks in the field, (2) provide students opportunities to draw from their own diverse backgrounds and experiences, (3) associate built-in evaluation and feedback. Examples can include application tasks specific to the discipline.

Therefore, the design of learning units involves selecting appropriate activities from each instructional phase, taking into account the necessary support and practice learners will need to grasp a particular concept and achieve the associated learning outcome.

4.1 System Design and Implementation. The system workflow design for planning iVR-based learning units comprises of three stages: (A) Developing a library of learning activities, (B) designing the instructional flow using a web-based editor, and (C) testing and verifying the instruction sequence of the learning activities. Before going into the details of each stage, we would like to mention that the current system is built on our previous work [30]. The current system differs from the previous stage, particularly in Stage B where the web-based editor is provided with LLM-supported interactive capabilities with an aim to make the lesson planning easier for the user. The interaction is guided by following the principles of Backward design and thus aligns the learning units based on the target learning outcomes. Nevertheless, we would still go through the different stages to provide a comprehensive understanding to the user.

Developing a library of learning activities: The first step involves SMEs and VR developers collaboratively framing learning activities by designing backward from the overarching learning outcomes. VR developers then implement these activities while saving JSON configuration files that capture their properties. These files allow SMEs to later review and modify parameters, enabling personalization of training scenarios.

To validate our approach, a library of 27 reusable learning activities was developed for the welding use case. The activities were organized under the four instructional phases of Introduction, Presentation, Practice, and Application, and collectively covered four instructional topics: (1) MIG welding safety, (2) welding equipment, (3) welding wire-change maintenance, and (4) Tee-joint welding. For example, learners may first be introduced to Tee-joint welding and welding safety concepts, followed by instructional demonstrations of welding equipment and welding procedures. Learners then engage in practice activities such as identifying personal protective equipment and performing guided virtual welding exercises before completing authentic application tasks such as Tee-joint welding and welding wire-change maintenance.

Examples of activities included *intro_mig_safety* (introduction to welding safety concepts), *presentation_mig_safety* (video-based instruction on workplace safety and personal protective equipment), *practice_ppe* (identification of personal protective equipment), *practice_virtual_welding* (guided Tee-joint welding practice), and *application_tee_joint* (authentic welding task performance on a truck tailgate). A few examples of the learning activities are shown in Table 1.

The activity design was informed by our prior work with SMEs, where we applied Backward Design principles from learning sciences to ensure learners could achieve the intended outcomes [57]. These activities served as reusable building blocks that could be selected, reordered, and connected by subject matter experts to construct training scenarios aligned with desired learning outcomes. An example lesson plan generated using these activities is shown in Figure 2, illustrating how individual activities can be combined into a complete instructional sequence.

Designing the instructional flow using a graph-based flow editor: To create the sequence of the learning activities catering to the needs of training, we developed a web-based flow editor as shown in Figure 1. The web-editor comprises of two parts: (A) the interactive user interface on the left section and (B) the graph-based editor on the right section. An example implementation is shown in Figure 2.

In the web-based editor, nodes represent learning activities from the library, which can be placed on the interface by selecting the desired activities. SMEs may add any number of nodes to construct their training scenarios, connecting them with directed edges to define the instructional flow. A resource library tab organizes available materials into four categories: Introduction, Presentation, Practice, and Application. The interface also includes a *Controls* button with detailed usage instructions, along with *Upload*, *Delete* and *Download* options for managing entire lesson plans. Users may also delete individual nodes or edges directly. Selecting a node and pressing the 'Y' key opens a dialog box showing the node details, including editable properties. These properties allow SMEs to personalize activities by configuring: (1) *Timing:* (duration of the activity), (2) *Message:* (short description or learning objective shown to the learner), and (3) *Hint:* (whether to display audio/visual cues). Nodes are color-coded by category for clarity, and built-in prompts check for issues such as multiple incoming/outgoing edges or conflicting sequences.

The user interface on the left guides users to interactively provide their requirements for the lesson plans. The process starts with providing the learning outcomes, then defining the objectives, skills, assessment criteria and finally the learning activities to achieve the outcomes. The interface is supported by LLM-based capabilities that can update the different parameters of the planning process based on the user's inputs and in a hierarchical manner. For example, updating the learning objectives updates the following parameters in the hierarchy, i.e., skills, assessment and learning activities. After the sequence of learning activities gets generated, the user can generate the lesson graph that renders the sequence on the graph-editor on the right. Users can iterate upon the lesson plan using the functionalities of the graph-based editor till the target requirements are achieved.

The idea behind the prompting guidance is illustrated in details in Figure 3. The figure represents the structured logic used to drive the LLM-assistive lesson-plan authoring interface. It shows how each stage of the planning process, i.e., learning outcomes, objectives, skills, and assessment criteria, is hierarchically linked so that updates at one level cascade to the others. Each list (objectives, skills, or criteria containing three items by default) is explicitly aligned with the level above it: criteria $\rightarrow$skills $\rightarrow$objectives $\rightarrow$outcomes. This structured prompting ensures coherence and pedagogical consistency in the model's outputs by embedding the principles of Backward Design into the LLM workflow. Such hierarchical prompting schema helps the model generate measurable, outcome-aligned components of a lesson plan in a transparent and iterative way. This process enables instructors or subject-matter experts to build and refine instructional plans with reduced cognitive load and maximum alignment to desired learning outcomes.

Testing and verifying the instruction sequence of the learning activities: Once the scenario flow is designed, SMEs can save the corresponding JSON file in a designated path. When launched, the Unity front end reads this file to dynamically generate buttons labeled with the node names on the instruction screen. A Depth First Search (DFS) algorithm determines the sequence in which the learning activities appear. SMEs may either follow the activities in order or jump directly to a specific activity by selecting its button. After reviewing the sequence, they can return to the editor to revise the lesson plan until the scenario design is finalized.

5 User Study

5.1 Study Setup. The study was conducted with 8 participants (1 female, 7 male)(Age Range (number of users)=25-34 (8)) with prior welding experience who were recruited by word of mouth, email and flyer distribution. All users had prior experience with welding. 7 users had received prior training experience (informal or formal) in welding. Six users were novices in development for VR applications, two users rated themselves as intermediate in developing VR content. Five users had prior teaching experience, from which four of them designed curricula or learning instructions. Each user study was divided into three sessions where the task was to build six training scenarios, two per session using two systems SystemA and SystemB. SystemA is the graph-based editor without the interactive capabilities and is used as the baseline condition. SystemB is the FlowTrainer system with LLM-based capabilities. Each user was asked to plan the scenarios for three levels of task complexities in order using both systems. The three levels of task complexity were targeted towards gradually varying learning objectives. During this three-session study, the usability and mental workload of SystemB was evaluated while examining how the varying levels of task complexity affected the task performance and flexible authoring of lesson plans. Furthermore, the results were also compared with the baseline condition SystemA to gain insights on the perceived user preferences. The study was approved under the IRB protocols.

5.2 Study Procedure. Users were first required to sign the consent forms and then proceeded to complete a short demographics survey. User demographics was collected about any prior experience with welding, VR, VR development, curriculum or instruction design, and teaching. Next, users were provided with the basic information in the form of a Word document and a PowerPoint presentation explaining the overall theme of the study. The

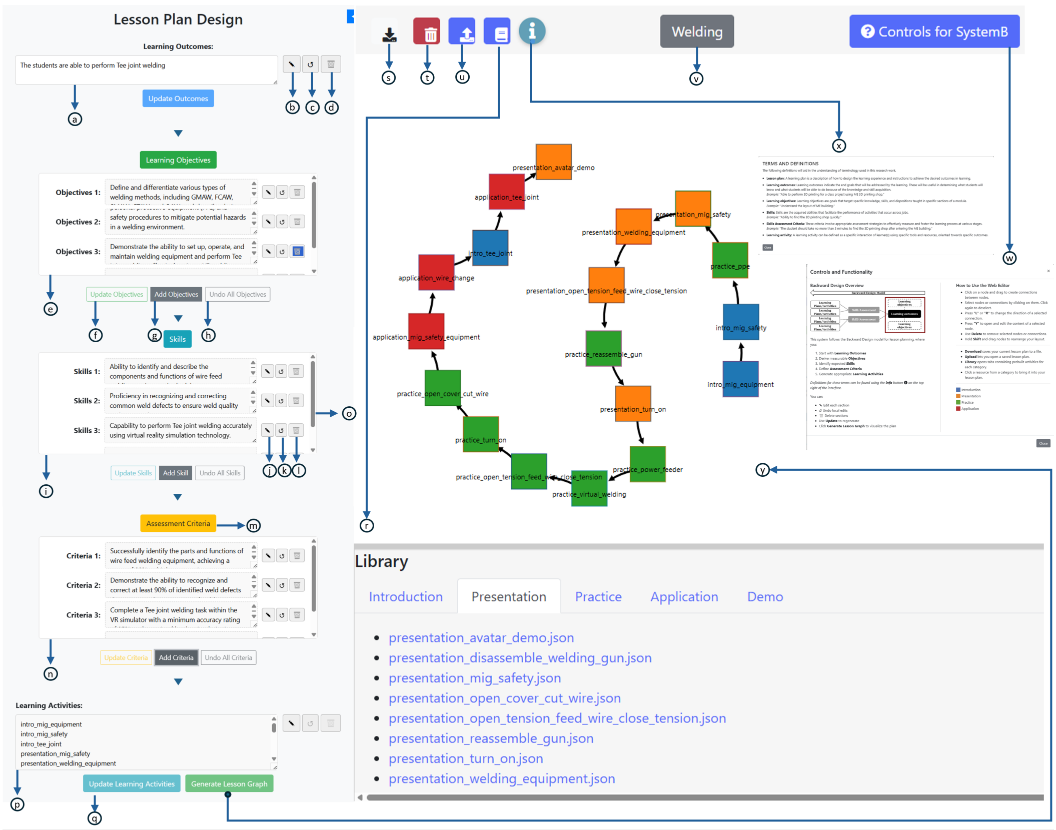


Fig. 1 **System Features of FlowTrainer: The left section of the UI presents an interface that assists user to generate a lesson plan by following the different stages of Backward Design. (a) The user can enter the learning outcomes, with features for (b) Edit, (c) Undo, and (d) Delete. The system generates (e) three learning objectives, (i) three skills and (n) three assessment criteria to measure the skills. Based on these a set of (p) learning activities are generated by the system. Local Edit, Undo and Delete features are used for individual objectives, skills or criteria (e.g. as shown in (j), (k), and (l)) where as there are global features such as add, delete and update objectives, skills and assessment criteria. The update of the hierarchical items happen in an hierarchical way, i.e., updating objectives have precedence over skills update however has lower precedence over updating outcomes. The learning activities can also be manually updated using (q) by adding or deleting learning activities. Collapsible buttons such as (m) and scroll bar such as (o) help in accessing the UI content. The lesson graph gets generated by using the button Generate Lesson graph which renders the lesson plan (graph sequence) on the web editor in the right section of the UI as shown in (y). The web editor has functionalities to edit the lesson graph using the resources from the library tab which can be opened by the Library button as shown in r. The user can click on any learning activities to bring the nodes to the web editor as required and create connections to add appropriately to the sequence. The user can save, delete or upload lesson plans using the buttons (s), (t), and (u) respectively. The details about the instructions on how to use the interface can be found in the modal instruction box which is accessed using the (w). Some definitions of the terminologies can be accessed using the modal instruction box (x). The button (v) at the top of the editor switches the mode between Demo and Welding for the user study.**

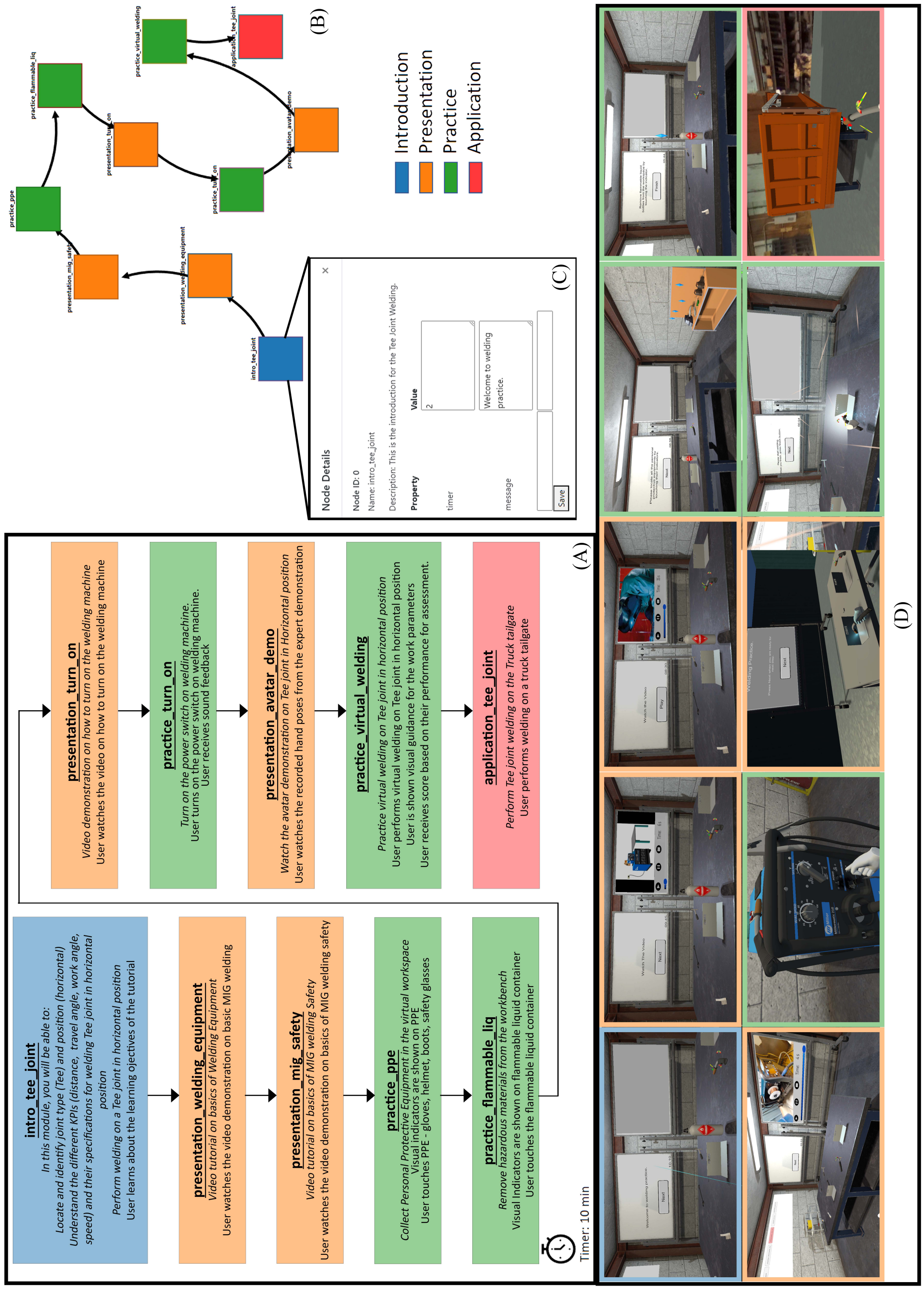


Fig. 2 **An example implementation of the instructional planning of learning units using FlowTrainer: (A) Flowchart representation of the step by step planning of the learning unit, (B) Authoring the sequence of the iVR learning unit using the web-based editor, (C) The dialog prompt showing the details and editable properties of the node, and (D) Verification and testing of the learning unit in VR**

Prompt to generate activity sequence based on outcomes, objectives, skills, criteria and context

```
def generate_activity_sequence(learning_outcomes, pdf_text,
      activities, lesson_plan):
  """Use LLM to determine the sequence of multiple learning
      activities across all categories."""

  prompt = f"""
  Given the following:

      - Learning outcomes: "{learning_outcomes}"
  - Objectives: {json.dumps(lesson_plan['objectives'],
      indent=2)}
  - Skills to be taught: {json.dumps(lesson_plan['skills'],
      indent=2)}
  - Skill assessment criteria:
      {json.dumps(lesson_plan['criteria'], indent=2)}

      Context from a teaching resource document:
  {pdf_text}

      Available learning activities organized by category:
  {json.dumps(activities, indent=2)}

      Select and sequence the most relevant activities (you
      may choose multiple) from each category:
  - Introduction
  - Presentation
  - Practice
  - Application

      Choose those that best support the objectives and skill
      development.

  Output a JSON list of activity names in the final instructional
      order.
  Example:
  {{
   "sequence": [
    "Intro Activity 1",
    "Intro Activity 2",
    "Presentation Activity 1",
    "Practice Activity 1",
    "Practice Activity 2",
    "Application Activity 1"
   ]
  }}
      """
```

Inputs
learning_outcomes: str
pdf_text: str
activities: dict[str, list[str]]
Keys: "Introduction" | "Presentation" | "Practice" | "Application"
lesson_plan: dict
Keys: objectives: list[str], skills: list[str], criteria: list[str]

Output
{"sequence": list[str]} - an ordered list of activity names aligned to *skills←objectives←outcomes.*

Prompt to generate objectives, skills, criteria based on outcomes and context

```
def generate_lesson_plan(learning_outcomes,
    pdf_text):
 """Use LLM to generate objectives, skills, and
    criteria based on learning outcomes and PDF
    context."""

 prompt = f"""
 Given the following learning outcomes:
    "{learning_outcomes}", and the contextual
    content from a teaching PDF resource:

 {pdf_text}

 Based on this information, generate:
 - Three specific and measurable learning objectives
 - Three skills that learners are expected to gain
 - Three skills assessment criteria to measure the
    skills and learning

 Format the output as JSON:
 {{
  "objectives": ["Objective 1", "Objective 2",
    "Objective 3"],
  "skills": ["Skill 1", "Skill 2", "Skill 3"],
  "criteria": ["Criteria 1", "Criteria 2", "Criteria 3"]
 }}
          """
```

Inputs
learning_outcomes: str
pdf_text: str

Output
{"objectives": list[str], "skills": list[str], "criteria": list[str]}
Each list has 3 items, that are aligned to *outcomes.*

Prompt to generate skills based on outcomes, objectives and criteria and context

```
def regenerate_skills(objectives, learning_outcomes,
    pdf_text):
 """Use LLM to generate skills and assessment
    criteria based on outcomes and objectives."""

 prompt = f"""
 Given the following:
 - Learning outcomes: "{learning_outcomes}"
 - Objectives: {json.dumps(objectives, indent=2)}
 - Context from a teaching resource document:

     {pdf_text}

     Based on this information, generate:
 - Three skills aligned with the objectives
 - Three skills assessment criteria to measure the
    skills and learning

     Format the output as JSON:
 {{
  "skills": ["Skill 1", "Skill 2", "Skill 3"],
  "criteria": ["Criteria 1", "Criteria 2", "Criteria 3"]
 }}
          """
```

Inputs
objectives: list[str] (length ≥ 1)
learning_outcomes: str
pdf_text: str

Output
{"skills": list[str], "criteria": list[str]}
Each list has 3 items, that are aligned to *objectives←outcomes.*

Prompt to generate criteria based on outcomes, objectives, skills and context

```
def regenerate_criteria(objectives, skills,
    learning_outcomes, pdf_text):
 """Use LLM to generate skill assessment criteria
    based on outcomes, objectives and skills."""

 prompt = f"""
 Given the following:
 - Learning outcomes: "{learning_outcomes}"
 - Objectives: {json.dumps(objectives, indent=2)}
 - Skills: {json.dumps(skills, indent=2)}
 - Context from a teaching resource document:

     {pdf_text}

     Based on this information, generate:
 - Three skills assessment criteria to measure the
    skills and learning

     Format the output as JSON:
 {{
  "criteria": ["Criteria 1", "Criteria 2", "Criteria 3"]
 }}
          """
```

Inputs
objectives: list[str] (length≥1)
skills: list[str] (length≥1)
learning_outcomes: str
pdf_text: str

Output
{"criteria": list[str]} - The list has 3 items, that are aligned to *skills←objectives←outcomes.*

Fig. 3 Prompting logic for the LLM-assistive user interface to author lesson plans

Table 1 Representative learning activities available in the welding activity library.

Instructional Phase	Example Activity	Description
Introduction	intro_mig_safety	Introduces welding workplace safety concepts.
Introduction	intro_tee_joint	Introduces Tee-joint welding and learning objectives.
Presentation	presentation_mig_safety	Video-based instruction on safety rules, PPE, and hazards.
Presentation	presentation_avatar_demo	Avatar demonstration of Tee-joint welding procedure.
Practice	practice_ppe	Identification and collection of PPE in the virtual workspace.
Practice	practice_virtual_welding	Guided virtual welding practice with feedback.
Application	application_tee_joint	Performance of a complete Tee-joint welding task.
Application	application_wire_change	Application of welding wire-change maintenance procedure.

information included the objective of the study, a short description about the four instructional phases, and the definitions of the stages involved in the Backward design lesson planning.

To get accustomed to the system functionalities, users were asked to interact with the SystemA and SystemB using a cooking demo. For the cooking demo, they were asked to design a tutorial for pizza baking. The cooking demo utilized a demo activity library consisting of 16 learning activities distributed equally across the four instructional phases of Introduction, Presentation, Practice, and Application (See Table 4 in Appendix). Representative examples included activities such as *intro_pizza_types*, *presentation_dough_steps*, *practice_dough_kneading*, and *application_bake_and_serve*. The available resources in the activity library were enumerated in a table with their names and a short description, the sequence of which was randomly chosen. Users were asked to come up with a tutorial title, write the learning outcomes on the sheet, and then design the lesson plan using both systems, SystemA followed by SystemB.

The cooking activity library was used solely to familiarize participants to the flow-editor interface. Unlike the welding use case, corresponding iVR learning activities were not developed for the cooking domain. Participants interacted only with the activity descriptions and authoring interface during this demo.

After the demo, the actual study process for the welding simulator began. Users were asked to design lesson plans for the welding simulator. The resource library provided to participants consisted of 27 learning activities. The library consisted of 4 Introduction activities, 9 Presentation activities, 10 Practice activities, and 4 Application activities. Collectively, these activities covered MIG welding safety, welding equipment, welding wire-change maintenance, and Tee-joint welding (See Table 5 in Appendix). The available resources were enumerated in a table with their names and a short description, the sequence of which was randomly chosen. Participants were free to select any subset of these activities and organize them into instructional sequences based on the specified learning objectives. The PowerPoint presentation was prepared containing visual description of the available iVR-based learning activities, in the form of videos.

After going through the resources, users were informed about the task of creating three training scenarios in three different sessions in order using SystemA and SystemB, the requirements of which were gradually disclosed immediately before the corresponding session started. Users were asked to consult the researcher present in the scene in case of any questions during the study. The order in which the user utilized the systems was counterbalanced to allow for within subjects comparision, while countering any order effects (e.g. learning, fatigue, etc.). The ordering resulted in the decision to include 8 users for the study. Each session had two tasks, during which the user performed the lesson planning using either SystemA or SystemB. Each lesson planning ended with a short survey to collect user ratings on the System Usability Scores (SUS) and workload (NASA-TLX) associated with the planning process using the system.

As shown in Table 2, the study was structured around three progressively evolving instructional design sessions. This progressive modification of learning objectives was intended to evaluate the ability of participants to adapt instructional sequences using the proposed workflow.

For the first session, users were first informed to imagine themselves to be the welding instructor of a class. The task was to plan a Tee joint Welding tutorial for a VR welding simulator to teach the imaginary class of students. Users were asked to come up with a tutorial title and the associated learning outcomes on the instruction document. This followed with the users designing the tutorial using the systems, one after another in the order pre-defined for the user. Each tutorial planning ended up with survey that collected scores for system usability and mental workload about the particular system.

During the second session, the users were informed about a change in requirements from the first session that the students in the imaginary class were novices in the field and did not have prior experience in welding. Considering this change in requirement, the new task was to build upon the previous tutorial plan to include additional information on basics of MIG welding, e.g., welding safety and equipment. The study procedure remained the same as before where users modified the previous tutorial plan based on the new information using both systems.

During the third session, the requirement changed again. It was realized that the students were needed to be taught about the welding wire change maintenance, and thus the tutorial had to be modified based on the updated requirements. Same as before, the users were asked to modify the tutorials using both systems.

After the three sessions were finished, the users were asked to finish a survey questionnaire asking them to record their responses about their preferences with using both systems. The survey also asked subjective responses regarding what features they liked or disliked in both systems and suggestions for improvement. This feedback was later used in the paper to explain the study results and inspire future design insights. The study lasted for approximately 1.5 hours and each user was given $15 compensation.

Data was collected for each user in terms of: (i) time required to finish each session task, (ii) JSON files for the training scenarios, (iii) 5-point Likert scale-based questionnaires (e.g., SUS, and NASA-TLX after each task, and a custom questionnaire after the three sessions were finished), and (iv) subjective feedback collected at the end of the study.

A researcher recorded the duration of the tasks in real time during the three sessions. Each user study session was recorded by using a screen capture, in case a review was needed or timing was missed. Data about the welding training scenarios was recorded from the flow editor for the three sessions. This data was analyzed later to compare performance metrics of both systems.

Table 2 Instructional design sessions during the user study.

Session	Task Description
1	Design a lesson plan for Tee-joint welding.
2	Modify the lesson plan to include MIG welding safety and equipment for novice learners.
3	Further revise the lesson plan to include welding wire-change maintenance.

The JSON files from the training scenarios were analyzed to collect the sequence length distributions and analyze instructional sequence structures using node-link diagrams across the different sessions. Sequence length was defined as the number of learning activities included in a lesson plan and was used as an indicator of lesson-plan adaptation. In the node-link diagrams as shown in Figure 5, nodes represent learning activities, while directed edges represent transitions between activities within a lesson plan. The size of each node is proportional to its frequency of occurrence across participants, and the thickness of each edge indicates the frequency with which a particular transition occurred. This data was analyzed later to better understand how participants adapted lesson plans across the three study sessions using both systems.

5.3 Results. The quantitative results of the study provide a comprehensive comparison of SystemA and SystemB across usability, workload, and performance metrics in terms of timing and sequence length. After checking that the assumptions of normality (Shapiro-Wilk test, p >0.05) were not met, non-parametric methods were utilized to find any statistical significance in the results. The survey results are described while providing the scores for average (Avg) and standard deviation (SD).

System Usability Scores for SystemA and SystemB across sessions: System Usability Scale (SUS) scores were computed for each session, with SystemB showing higher usability ratings than SystemA during Sessions 1 and 3. In Session 1, SystemB had an average SUS score of 74.38 (SD = 8.53) compared to SystemA with an average of 72.19 (SD = 12.35). In Session 2, SystemB scored 75.00 (SD = 13.36) while SystemA slightly surpassed it with 76.56 (SD = 13.82). However, in Session 3, SystemB had a significant advantage with a score of 83.12 (SD = 10.42), compared to SystemA with an average of 73.44 (SD = 19.64).

Statistical analysis using the Friedman test showed that task complexity did not significantly impact usability ratings for either system. For SystemA, the test yielded χ^2 = 1.31, p = 0.5194, and for SystemB, χ^2 = 4.22, p = 0.1211. Post-hoc Wilcoxon tests also showed no significant pairwise differences across complexity levels for either system after Bonferroni correction. For SystemA, all comparisons (e.g., Session1 vs. Session2, Session2 vs. Session3) had p-values > 0.52, and for SystemB, while some comparisons approached significance (e.g., Session1 vs. Session3 at p = 0.093), none met the corrected threshold (p = 0.0167). The box plots showing the SUS scores can be found in Figure 4(a).

NASA-TLX ratings for SystemA and SystemB across Sessions: The NASA-TLX results across the three sessions reveal how SystemA and SystemB compare in terms of perceived workload. The results can be found in can be found in Figure 4(b-f).

In Session 1, both systems exhibited comparable mental workload, with SystemA averaging 2.75 (SD=3), and SystemB slightly higher at 2.88 (SD=3), though variability was lower in SystemB. Physical and temporal demands were low overall, while effort was slightly greater for System with an average of 3.75 (SD=4), compared to SystemA with an average of 3.63 (SD=4). Performance ratings remained equal with an average of 2.5 (SD=3), indicating similar perceived effectiveness. In Session 2, SystemA had slightly lower mental demand (Avg=2.00, SD=2) than SystemB (Avg=2.13, SD=2), but required more effort in SystemA (Avg=4.25, SD=4) as compared to SystemB (Avg=3.63, SD=4). Physical and temporal demands were identical across systems. In Session 3, SystemB outperformed SystemA, with lower averages across all workload dimensions, mental demand dropped in SystemB (Avg=1.5, SD=2) as compared to SystemA (Avg=2.25, SD=2), and effort was significantly lower in SystemB (Avg=2.5, SD=3) as compared to SystemA (Avg=3.75, SD=4). These values indicate that as task complexity increased, SystemB was perceived as less mentally and physically demanding.

Statistical analyses further confirm these observations. Friedman tests showed that task complexity had a significant effect on mental workload in SystemB (χ^2 = 11.27, p = 0.0036), but not in SystemA (p = 0.0578). Post-hoc Wilcoxon tests revealed a significant difference in SystemB between Session1 and Session3 (p = 0.0158), whereas SystemA showed no significant pairwise differences. For physical workload, the effect of task complexity was significant in SystemA (χ^2 = 6.13, p = 0.0468) but not in SystemB (p = 0.1738), although post-hoc tests revealed no significant pairwise differences in either system after Bonferroni correction. Neither system showed significant effects of task complexity on temporal demand, performance, or effort based on Friedman test results (all p = 0.05). Furthermore, all post-hoc comparisons across these metrics failed to reach significance, reinforcing the conclusion that only mental workload in SystemB and physical workload in SystemA were impacted by increasing task complexity.

These findings demonstrate that SystemB maintains lower and more consistent workload under complexity, particularly in terms of cognitive load. While SystemA was more sensitive to task complexity in physical demand, it did not lead to meaningful differences in user performance or overall effort. This supports the suitability of SysytemB for flexible, scalable use in authoring VR training content, especially for more cognitively demanding scenarios.

Time of Task Completion for SystemA and SystemB across sessions: In terms of task completion time, SystemA showed a noticeable drop from Session 1 to Sessions 2 and 3, with average timing decreasing from 7.19 seconds (SD = 4.13) to 2.39 (SD = 1.79) and 2.79 seconds (SD = 1.28), respectively. SystemB also showed a decrease, starting from 4.94 seconds (SD = 3.48) in Session 1 and dropping to 2.98 (SD = 1.44) in Session 2 and 1.71 seconds (SD = 0.77) in Session 3. These results suggest that users required less time as they became more familiar with the systems or as the task types changed. However, Friedman tests revealed no statistically significant effect of task complexity on timing for either system (SystemA:χ^2 = 4.75, p = 0.0930; SystemB:χ^2 = 5.25, p = 0.0724), and the Wilcoxon test showed no significant difference in timing variability between the systems (p = 0.3828). Post-hoc tests further confirmed that none of the pairwise comparisons across task complexities reached significance under Bonferroni correction. The results can be found in can be found in Figure 4(g).

Sequence Length Distributions for SystemA and SystemB across Sessions: Regarding the sequence length distribution, SystemA increased from 16.00 (SD = 4.93) in Session 1 to 18.62 (SD = 5.01) in Session 3, while SystemB increased from 14.62 (SD = 5.95) to 20.12 (SD = 2.95). Friedman tests revealed a significant effect of task complexity on sequence length for both systems (SystemA:χ^2 = 10.57, p = 0.0051; SystemB:χ^2 = 6.07, p = 0.0482). However, the Wilcoxon test found no significant difference between systems regarding how much complexity affected sequence length (p = 0.4990). In post-hoc analysis, only SystemA showed a significant difference between moderate and complex tasks (p = 0.0078. In SystemB, no pairwise comparison reached significance under Bonferroni correction. Additionally, the Wilcoxon test comparing variability in sequence length between systems found no significant difference (p = 1.000). The results can be found in can be found in Figure 4(h).

Analysis of Instructional Sequence Structures: The authored instructional sequences from the study were analyzed using node-

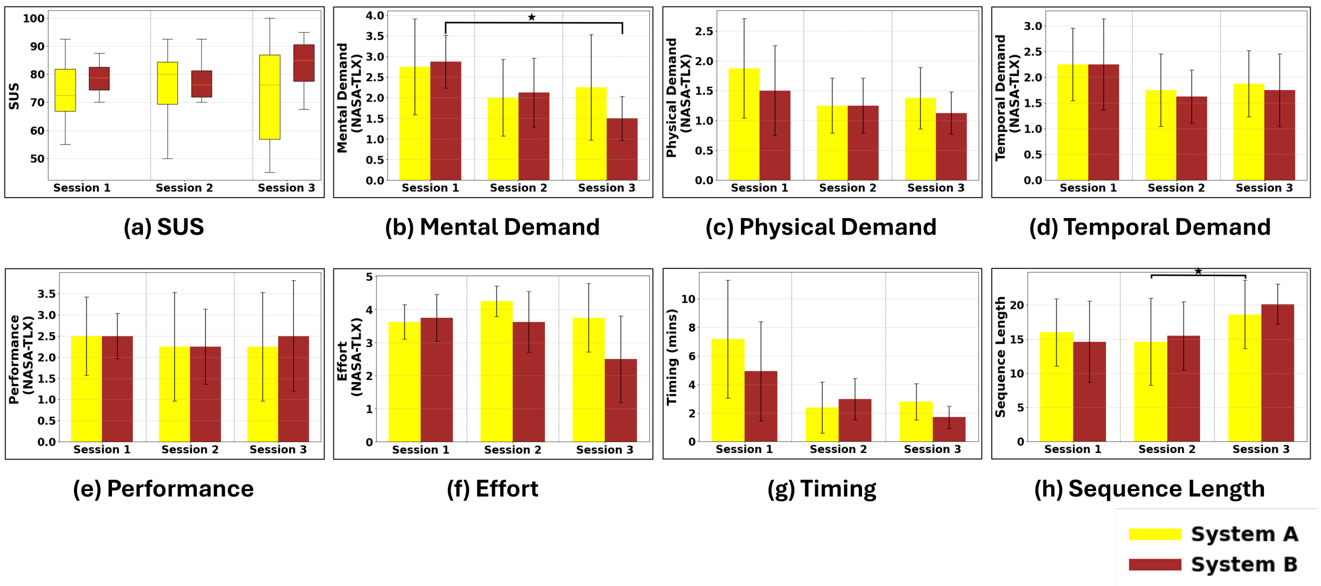


Fig. 4 Quantitative Results from the Survey Questionnaire showing the comparision between SystemA (web-based editor without LLM capabilities) and SystemB (web-based editor with LLM capabilities): (a) System Usability Scores (SUS) across the three sessions, (b-f) NASA-TLX ratings for Mental Demand, Physical Demand, Temporal Demand, Performance and Effort across the three sessions, (g) Time of Completion of tasks across the three sessions, and (h) Sequence Length Distribution across the three sessions.

link diagrams, as shown in Figure 5. Visual inspection of the diagrams reveals changes in activity selection and sequencing as the instructional requirements evolved across sessions. Session 1 primarily focused on Tee-joint welding, whereas Session 2 introduced additional objectives related to MIG welding safety and equipment. Session 3 further incorporated welding wire-change maintenance. As a result, activities associated with these topics became increasingly prevalent in the authored lesson plans. The diagrams also show recurring transitions between activities belonging to the Introduction, Presentation, Practice, and Application phases, suggesting that participants generally followed the intended instructional structure when constructing lesson plans. The accompanying tables summarize the most frequently occurring nodes and edges for each session.

Custom questionnaire on User Preferences: The final survey questionnaire results are summarized in Table 3. From the questionnaire on a 1-5 Likert Scale *[1=Strongly Disagree, 5=Strongly Agree]*, the mean and standard deviation values were calculated. Average user ratings less than 3 suggests stronger preferences for SystemB compared to SystemA and vice versa. It was observed that participants generally leaned toward slightly negative to neutral responses on most statements comparing SystemA to SystemB. They somewhat disagreed that SystemA was simpler to use (Avg=2.43, SD=1.05) and easier to get familiar with (Avg=2.57, SD=1.4). Opinions were slightly negative regarding whether SystemA's interface better captured their intent (Avg=2.71, SD=1.03) and whether it helped with tutorial planning (Avg=2.86, SD=0.99). Users slightly agreed that SystemA allowed easier modification of tutorials based on changing requirements (Avg=3.43, SD=1.29) and iteration on existing plans (Avg=3.71, SD=1.16). They somewhat disagreed that SystemA required a lot of preparation (Avg=2.43, SD=1.18). Ratings were neutral on SystemA's usefulness for creating VR lessons (Avg=3.00, SD=1.51) and slightly below neutral for instructors using it in classrooms (Avg=2.71, SD=1.03).

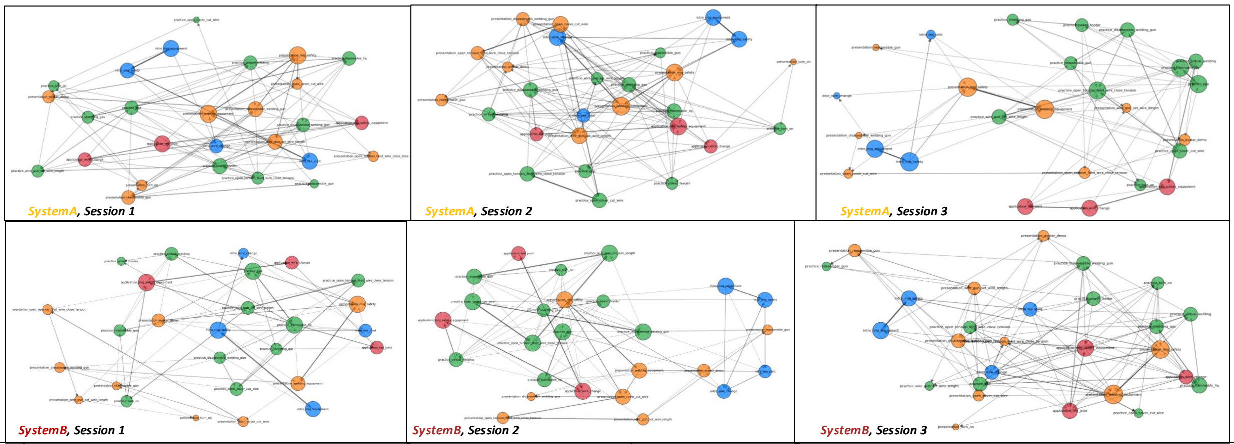
6 Discussion

Based on the results and subjective feedback from users, we now try to answer the research question that we posed before. ***To what extent does the system support SMEs in authoring and adapting training scenarios based on varied learning objectives as compared to the baseline condition?***

Evidence for flexible authoring can be observed from the manner in which participants successfully adapted lesson plans across the three study sessions. Rather than creating independent lesson plans, participants were required to progressively modify existing plans in response to evolving instructional requirements. Session 1 focused on Tee-joint welding, Session 2 introduced additional learning objectives related to MIG welding safety and equipment, and Session 3 further incorporated welding wire-change maintenance. These changes required participants to revisit previously authored lesson plans, select additional learning activities, modify instructional sequences, and reorganize the flow of instruction.

The sequence-length analysis provides further insight into authoring flexibility. As new learning objectives were introduced across sessions, participants expanded their lesson plans by incorporating additional learning activities and instructional steps. This trend was observed in both systems, where sequence lengths generally increased with task complexity. The increasing sequence lengths indicate that participants actively adapted lesson plans in response to changing instructional requirements rather than relying on fixed instructional structures. [*Note: In the context of this research, sequence length should not be interpreted as inherently better or worse. Rather, it serves as an indicator of the extent to which participants modified and extended lesson plans to accommodate new learning objectives.*]

The node-link analysis shown in Figure 5 provides additional insight into the flexible authoring capabilities of the proposed workflow. As the instructional requirements evolved across the three study sessions, participants modified their lesson plans by introducing new learning activities and connections between activities. The frequently occurring nodes and edges in each session correspond directly to the learning objectives introduced during that session. For example, activities related to MIG welding safety and equipment appeared frequently when novice learners were introduced in Session 2, while wire-change maintenance activities became more prominent in Session 3. These patterns indicate that participants were able to selectively reuse, reorganize, and extend learning activities to construct lesson plans aligned with changing instructional requirements.



	SystemA		SystemB	
Session	Nodes	Edges	Nodes	Edges
Session 1	presentation_welding_equipment, 8 presentation_mig_safety, 8 intro_mig_equipment, 7 intro_tee_joint, 7 application_tee_joint, 7	('intro_mig_equipment', 'intro_mig_safety'), 4 ('presentation_welding_equipment', 'presentation_mig_safety'), 3 ('intro_tee_joint', 'presentation_welding_equipment'), 2 ('presentation_mig_safety', 'presentation_wire_gun_set_wire_length'), 2 ('presentation_avatar_demo', 'presentation_disassemble_welding_gun'), 2	intro_mig_safety, 8 presentation_mig_safety, 8 practice_flammable_liq, 8 practice_ppe, 8 application_mig_safety_equipment, 8	('intro_mig_equipment', 'intro_mig_safety'), 6 ('practice_open_tension_feed_wire_close_tension', 'practice_wire_gun_set_wire_length'), 5 ('application_mig_safety_equipment', 'application_wire_change'), 5 ('presentation_mig_safety', 'presentation_welding_equipment'), 4 ('practice_ppe', 'practice_flammable_liq'), 4
Session 2	presentation_welding_equipment, 9 presentation_mig_safety, 8 intro_mig_equipment, 7 application_mig_safety_equipment, 7 intro_mig_safety, 6	('intro_mig_equipment', 'intro_mig_safety'), 4 ('presentation_mig_safety', 'practice_ppe'), 3 ('presentation_mig_safety', 'presentation_welding_equipment'), 3 ('intro_wire_change', 'presentation_welding_equipment'), 2 ('presentation_welding_equipment', 'presentation_mig_safety'), 2	intro_mig_safety, 8 presentation_mig_safety, 8 presentation_welding_equipment, 8 intro_mig_equipment, 7 practice_flammable_liq, 7	('presentation_mig_safety', 'presentation_welding_equipment'), 5 ('intro_mig_equipment', 'intro_mig_safety'), 4 ('application_mig_safety_equipment', 'application_wire_change'), 4 ('practice_flammable_liq', 'practice_ppe'), 3 ('presentation_welding_equipment', 'presentation_mig_safety'), 3
Session 3	presentation_welding_equipment, 9 intro_wire_change, 8 presentation_mig_safety, 8 application_mig_safety_equipment, 8 intro_mig_equipment, 7	('intro_mig_equipment', 'intro_mig_safety'), 4 ('practice_open_cover_cut_wire', 'practice_open_tension_feed_wire_close_tension'), 4 ('intro_wire_change', 'presentation_welding_equipment'), 3 ('presentation_open_cover_cut_wire', 'presentation_open_tension_feed_wire_close_tension'), 3 ('presentation_mig_safety', 'presentation_welding_equipment'), 3	intro_mig_equipment, 8 presentation_mig_safety, 8 practice_disassemble_welding_gun, 8 practice_open_cover_cut_wire, 8 practice_open_tension_feed_wire_cl ose_tension, 8	('practice_open_cover_cut_wire', 'practice_open_tension_feed_wire_close_tension'), 6 ('practice_open_tension_feed_wire_close_tension', 'practice_wire_gun_set_wire_length'), 6 ('practice_wire_gun_set_wire_length', 'practice_reassemble_gun'), 6 ('practice_disassemble_welding_gun', 'practice_open_cover_cut_wire'), 5 ('presentation_open_cover_cut_wire', 'presentation_open_tension_feed_wire_close_tension'), 4

Fig. 5 Node link diagram for visualization of the graph sequences from the three-session user study for both systems alongwith the commonly recurring nodes and edges in the graph sequence

From the usability and cognitive load analysis, SystemB received consistently higher and more stable SUS scores especially in Sessions 1 and 3. This suggests that it provides a more user-friendly interface for content creation. The significant jump in usability scores during Session 3 (SystemB: 83.12 vs. SystemA: 73.44) suggests that SystemB scales better with the cognitive demands of complex tasks. As pointed out by a user, "*I can give a prompt, and it will make all the required steps, which if I feel needs to be changed or I might have missed, I can do that. SystemA is more time-consuming.*" Another user mentioned, "*It was easier to use, and I did less effort in typing out all the subsections of the lesson plan. It was better not to interact with the floating bubbles like in SystemA.*"

NASA-TLX results further support this conclusion. Mental workload in SystemB significantly decreased with increasing task complexity. This indicates that users found it cognitively easier to build complex training scenarios. On the other hand, increasing physical demand with complexity in SystemA may indicate limitations in its authoring affordances. As pointed out by one user, "*As the nodes grew and the arrows started crossing each other while editing, it was hard to keep track of everything.*" The need to repeatedly add and connect nodes manually even for straightforward scenarios, was mentioned as a drawback. One user suggested, "*simplify the input and edit mechanism to reduce all the bubbles with so many crossings to avoid confusion.*"

The trends for time of task completion showed that both systems improved with user familiarity. However, SystemB showed more efficient performance for complex tasks, with the shortest completion time of 1.71 minutes in Session 3, compared to 2.79 minutes in SystemA. This suggests that SystemB has a smoother learning curve. As pointed out by an user, "*It is easy to make the entire class presentation with a command and easy to modify.*"

The custom questionnaire provided additional insight into user perceptions of the two authoring systems. Participants generally perceived SystemB as simpler to use and easier to become familiar with. However, responses related to modifying tutorials under changing requirements and iterating on existing lesson plans slightly favored SystemA, potentially because it offers a more manual, hands-on workflow. One participant mentioned, "*The fact that I can choose the order in which I would like to plan the tutorials*" as a favorable aspect of SystemA, while also noting that "*It was cumbersome to know all the resources in A to plan the tutorials.*" This observation may suggest that while LLM-assisted guidance reduced effort during lesson-plan creation, some users preferred the direct manipulation afforded by the graph-based editor when revising existing instructional sequences.

For SystemB, users appreciated features such as "*automatic generation of tasks*" and "*lesson plan generation, setting the learning outcomes,*" though a few pointed out limitations, such as "*manual typing can be a bit hectic*" and "*the generated results looked similar in most cases.*" Future work should investigate how to better balance automation and user control during iterative lesson-plan refinement.

Participants successfully adapted lesson plans to changing instructional requirements using both systems, indicating that both authoring approaches supported the creation and modification of training scenarios based on varied learning objectives. However, the proposed workflow enabled participants to perform these adaptations with higher usability, lower workload, and reduced authoring effort compared to the baseline condition. These findings suggest that the LLM-assisted workflow better supports SMEs in authoring and adapting training scenarios under changing instructional requirements by reducing cognitive demand and improving ease of use. As one user put it, "*If there was a way to change the order of skills learning and practicing, then SystemB would be better than A.*"

Although the small sample size and the lack of experience in teaching and instruction for some users may affect the results, we have considered the counterbalancing effects for the comparative study. We acknowledge that this is a limitation in the study and the results may improve from a more targeted population and increased sample size. Nevertheless, the results presented in this work are still useful and can guide researchers in this field to facilitate the use of VR for learning purposes.

Table 3 Custom questionnaire results comparing SystemA with SystemB. Ratings were provided on a 5-point Likert scale *[1 = Strongly Disagree, 5 = Strongly Agree]*.

Survey statement	Mean	SD
SystemA was simpler to use than SystemB.	2.43	1.05
You could get familiar with SystemA more easily as compared to SystemB.	2.57	1.4
The interface of SystemA was more successful in capturing your intent for the flow of the tutorial as compared to SystemB.	2.71	1.03
SystemA helped you better in planning the tutorials for the welding simulator as compared to SystemB.	2.86	0.99
You could modify the tutorials as per the changing requirements in a better manner in SystemA as compared to SystemB.	3.43	1.29
You could iterate upon the existing plan to make desired changes in an easier way in SystemA as compared to SystemB.	3.71	1.16
SystemA can be more useful to create Virtual Reality based lessons for training purposes as compared to SystemB.	3.00	1.51
Instructors can utilize SystemA to create Virtual Reality based lessons for their classrooms in a better way as compared to SystemB.	2.71	1.03

7 Conclusion

We presented FlowTrainer, a workflow system that enables subject matter experts (SMEs) to plan iVR-based learning units aligned with training requirements. The workflow begins with VR developers creating a library of learning activities with configurable parameters, grounded in the Backward Design approach. These activities are organized under the instructional phases of Introduction, Presentation, Practice, and Application. This classification provides a scalable framework for lesson planning. Using these activities as building blocks, SMEs can design lesson plans through a graph-based editor supported by LLM-assisted interactions. Such LLM assistance streamlines authoring and help align plans with desired outcomes. A welding use case was implemented to validate the system. A user study with 8 experienced welders showed that FlowTrainer enabled SMEs to flexibly author training scenarios tailored to varied learning objectives. Participants reported positive usability and effectiveness ratings which underscores the potential of FlowTrainer to reduce barriers to iVR adoption. The results also showed that the FlowTrainer outperformed the baseline system in usability and workload. We believe these findings contribute to the broader adoption of immersive technologies in education and workforce training, thus offering a practical pathway for SMEs to design outcome-oriented VR learning experiences.

8 Acknowledgements

We thank the reviewers for their invaluable feedback. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the funding agency.

9 Funding

Not Applicable

10 Competing interests

The authors declare that they have no competing interests.

11 Data Availability Statement

Not applicable.

References

[1] Rose, E. L., 2021, "The decline of US manufacturing: Issues of measurement," Management and Organization Review, **17**(1), pp. 24–28.

[2] Al-Asfour, A. and Zhao, Y., 2024, "Bridging the skills gap divide in manufacturing: perspectives from industry leaders," Industrial and Commercial Training, **56**(1), pp. 78–90.

[3] Witt, M. A., Li, P. P., Välikangas, L., and Lewin, A. Y., 2021, "De-globalization and decoupling: game changing consequences?" Management and Organization Review, **17**(1), pp. 6–15.

[4] Al-Asfour, A., 2025, "Bridging manufacturing: Insights from industry leaders for enhanced human resource development," Advances in Developing Human Resources, **27**(2-3), pp. 42–61.

[5] Naranjo, J. E., Sanchez, D. G., Robalino-Lopez, A., Robalino-Lopez, P., Alarcon-Ortiz, A., and Garcia, M. V., 2020, "A scoping review on virtual reality-based industrial training," Applied Sciences, **10**(22), p. 8224.

[6] Badamasi, A. A., Aryal, K. R., Makarfi, U. U., and Dodo, M., 2022, "Drivers and barriers of virtual reality adoption in UK AEC industry," Engineering, Construction and Architectural Management, **29**(3), pp. 1307–1318.

[7] Scott, H., Baglee, D., O'Brien, R., and Potts, R., 2020, "An investigation of acceptance and e-readiness for the application of virtual reality and augmented reality technologies to maintenance training in the manufacturing industry," International Journal of Mechatronics and Manufacturing Systems, **13**(1), pp. 39–58.

[8] Jalo, H., Pirkkalainen, H., and Torro, O., 2021, *The State of Augmented Reality, Mixed Reality and Virtual Reality adoption and use in European small and medium-sized manufacturing companies in 2020: VAM Realities Survey Report*, VAM Realities Erasmus+ project.

[9] Fernandez, M., 2017, "Augmented virtual reality: How to improve education systems." Higher Learning Research Communications, **7**(1), pp. 1–15.

[10] Liagkou, V., Salmas, D., and Stylios, C., 2019, "Realizing virtual reality learning environment for industry 4.0," Procedia Cirp, **79**, pp. 712–717.

[11] Gaspar, H., Morgado, L., Mamede, H., Oliveira, T., Manjón, B., and Gütl, C., 2020, "Research priorities in immersive learning technology: the perspectives of the iLRN community," Virtual Reality, **24**, pp. 319–341.

[12] Coelho, H., Monteiro, P., Gonçalves, G., Melo, M., and Bessa, M., 2022, "Authoring tools for virtual reality experiences: a systematic review," Multimedia Tools and Applications, **81**(19), pp. 28037–28060.

[13] Walczak, K., Flotyński, J., Strugała, D., Strykowski, S., Sobociński, P., Gałązkiewicz, A., Górski, F., Buń, P., Zawadzki, P., Wielgus, M., et al., 2020, "Semantic modeling of virtual reality training scenarios," *Virtual Reality and Augmented Reality: 17th EuroVR International Conference, EuroVR 2020, Valencia, Spain, November 25–27, 2020, Proceedings 17*, Springer, pp. 128–148, doi: 10.1007/978-3-030-62655-6_8.

[14] Cassola, F., Pinto, M., Mendes, D., Morgado, L., Coelho, A., and Paredes, H., 2021, "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, pp. 44–49, doi: 10.1109/VRW52623.2021.00014.

[15] Ketoma, V. K., Vanderdonckt, J., and Meixner, G., 2023, "Towards Flexible Authoring and Personalization of Virtual Reality Applications for Training," Proceedings of the ACM on Human-Computer Interaction, **7**(EICS), pp. 1–37.

[16] Chang, Y.-h. I., Devine, K. L., and Klitzing, G. K., 2020, "Exploring the VR-based PBD programming approach to teach industrial robotics in manufacturing education," *2020 ASEE Virtual Annual Conference Content Access*, ASEE Conferences, Paper No. 10.18260/1-2–34650, https://peer.asee.org/34650.

[17] Yun, H. and Jun, M. B., 2022, "Immersive and interactive cyber-physical system (I2CPS) and virtual reality interface for human involved robotic manufacturing," Journal of Manufacturing Systems, **62**, pp. 234–248.

[18] Tram, M. Q., Cloud, J. M., and Beksi, W. J., 2023, "Intuitive Robot Integration via Virtual Reality Workspaces," *2023 IEEE International Conference on Robotics and Automation (ICRA)*, IEEE, pp. 11654–11660.

[19] Ipsita, A., Duan, R., Li, H., Chidambaram, S., Cao, Y., Liu, M., Quinn, A., and Ramani, K., 2024, "The Design of a Virtual Prototyping System for Authoring Interactive Virtual Reality Environments From Real-World Scans," Journal of Computing and Information Science in Engineering, **24**(3).

[20] Theofanidis, M., Sayed, S. I., Lioulemes, A., and Makedon, F., 2017, "Varm: Using virtual reality to program robotic manipulators," *Proceedings of the 10th International Conference on PErvasive Technologies Related to Assistive Environments*, pp. 215–221.

[21] Ipsita, A., Li, H., Duan, R., Cao, Y., Chidambaram, S., Liu, M., and Ramani, K., 2021, "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*, pp. 1–7.

[22] Seo, J., Kim, D., and Kim, G., 2001, "VR object reuse through component combination," *International Workshop on Structured Design of Virtual Environments and 3D-Components. Web3D Conference, Paderborn, Alemania*.
[23] Zikas, P., Papagiannakis, G., Lydatakis, N., Kateros, S., Ntoa, S., Adami, I., and Stephanidis, C., 2020, "Immersive visual scripting based on VR software design patterns for experiential training," The Visual Computer, **36**, pp. 1965–1977.
[24] McTighe, J. and Thomas, R. S., 2003, "Backward Design for Forward Action," Educational leadership, **60**(5), pp. 52–55.
[25] Wiggins, G. and McTighe, J., 1998, "What is backward design," Understanding by design, **1**, pp. 7–19.
[26] Wei, Y., Liu, H., Wu, H., and Chang, R., 2016, "OPME (Objectives-Process-Methods-Evaluation) Method for Lesson Planning," *2016 2nd International Conference on Social Science and Higher Education*, Atlantis Press, pp. 255–258, doi: 10.2991/icsshe-16.2016.66.
[27] Kruse, K., 2009, "Gagne's nine events of instruction: An introduction," https://psu.pb.unizin.org/, Accessed: 2026-8-9.
[28] Merrill, M. D., 2002, "First principles of instruction," Educational technology research and development, **50**, pp. 43–59.
[29] Bybee, R. W., Taylor, J. A., Gardner, A., Van Scotter, P., Powell, J. C., Westbrook, A., and Landes, N., 2006, "The BSCS 5E instructional model: Origins and effectiveness," Colorado Springs, Co: BSCS, **5**, pp. 88–98.
[30] Ipsita, A., Patel, M., Unmesh, A., and Ramani, K., 2024, "Authoring instructional flow in iVR learning units to promote outcome-oriented learning," Computers & Education: X Reality, **5**, p. 100074.
[31] Kaskalis, T. H., Tzidamis, T. D., and Margaritis, K., 2007, "Multimedia authoring tools: the quest for an educational package," Journal of Educational Technology & Society, **10**(3), pp. 135–162.
[32] Kumar, A., Saudagar, A. K. J., AlKhathami, M., Alsamani, B., Khan, M. B., Hasanat, M. H. A., and Kumar, A., 2022, "Customized Curriculum and Learning Approach Recommendation Techniques in Application of Virtual Reality in Medical Education." JUCS: Journal of Universal Computer Science, **28**(9).
[33] Zikas, P., Protopsaltis, A., Lydatakis, N., Kentros, M., Geronikolakis, S., Kateros, S., Kamarianakis, M., Evangelou, G., Filippidis, A., Grigoriou, E., et al., 2023, "MAGES 4.0: Accelerating the world's transition to VR training and democratizing the authoring of the medical metaverse," IEEE Computer Graphics and Applications, **43**(2), pp. 43–56.
[34] Coelho, H., Melo, M., Martins, J., and Bessa, M., 2019, "Collaborative immersive authoring tool for real-time creation of multisensory VR experiences," Multimedia Tools and Applications, **78**, pp. 19473–19493.
[35] Coelho, H., Melo, M., Barbosa, L., Martins, J., Teixeira, M. S., and Bessa, M., 2021, "Authoring tools for creating 360 multisensory videos—Evaluation of different interfaces," Expert Systems, **38**(5), p. e12418.
[36] Wolfartsberger, J. and Niedermayr, D., 2020, "Authoring-by-doing: Animating work instructions for industrial virtual reality learning environments," *2020 IEEE Conference on Virtual Reality and 3D User Interfaces Abstracts and Workshops (VRW)*, IEEE, pp. 173–176, doi: 10.1109/VRW50115.2020.00038.
[37] Fairén, M., Moyés, J., and Insa, E., 2020, "VR4Health: Personalized teaching and learning anatomy using VR," Journal of medical systems, **44**(5), p. 94.
[38] Cassola, F., Mendes, D., Pinto, M., Morgado, L., Costa, S., Anjos, L., Marques, D., Rosa, F., Maia, A., Tavares, H., et al., 2022, "Design and Evaluation of a Choreography-Based Virtual Reality Authoring Tool for Experiential Learning in Industrial Training," IEEE Transactions on Learning Technologies, **15**(5), pp. 526–539.
[39] Lécuyer, F., Gouranton, V., Lamercerie, A., Reuzeau, A., Caillaud, B., and Arnaldi, B., 2020, "Unveiling the implicit knowledge, one scenario at a time," The Visual Computer, **36**, pp. 1951–1963.
[40] Lugrin, J.-L., 2009, "Alternative reality and causality in virtual environments," PhD thesis.
[41] Mishra, P. and Koehler, M. J., 2006, "Technological pedagogical content knowledge: A framework for teacher knowledge," Teachers college record, **108**(6), pp. 1017–1054.
[42] Mahdi, O., Oubahssi, L., Piau-Toffolon, C., and Iksal, S., 2018, "Towards design and operationalization of pedagogical situations in the VRLEs," *2018 IEEE 18th International Conference on Advanced Learning Technologies (ICALT)*, IEEE, pp. 400–402, doi: 10.1109/ICALT.2018.00095.
[43] Saunier, J., Barange, M., Blandin, B., and Querrec, R., 2016, "A methodology for the design of pedagogically adaptable learning environments." International Journal of Virtual Reality, **16**(1), pp. 15–21.
[44] Kavanagh, S., Luxton-Reilly, A., Wuensche, B., and Plimmer, B., 2017, "A systematic review of virtual reality in education," Themes in Science and Technology Education, **10**(2), pp. 85–119.
[45] Keskitalo, T., 2011, "Teachers' conceptions and their approaches to teaching in virtual reality and simulation-based learning environments," Teachers and Teaching: theory and practice, **17**(1), pp. 131–147.
[46] Fan, H., Chen, G., Wang, X., and Peng, Z., 2024, "LessonPlanner: Assisting novice teachers to prepare pedagogy-driven lesson plans with large language models," *Proceedings of the 37th Annual ACM Symposium on User Interface Software and Technology*, pp. 1–20.
[47] Hu, B., Zhu, J., Pei, Y., and Gu, X., 2025, "Exploring the potential of LLM to enhance teaching plans through teaching simulation," npj Science of Learning, **10**(1), p. 7.
[48] Hu, B., Zheng, L., Zhu, J., Ding, L., Wang, Y., and Gu, X., 2024, "Teaching plan generation and evaluation with GPT-4: Unleashing the potential of LLM in instructional design," IEEE Transactions on Learning Technologies, **17**, pp. 1445–1459.
[49] Zheng, Y., Li, X., Huang, Y., Liang, Q., Guo, T., Hou, M., Gao, B., Tian, M., Liu, Z., and Luo, W., 2024, "Automatic Lesson Plan Generation via Large Language Models with Self-critique Prompting," *International Conference on Artificial Intelligence in Education*, Springer, pp. 163–178.
[50] Karpouzis, K., Pantazatos, D., Taouki, J., and Meli, K., 2024, "Tailoring education with GenAI: A new horizon in lesson planning," *2024 IEEE global engineering education conference (EDUCON)*, IEEE, pp. 1–10.
[51] Chen, Y., Yang, Y., and Xu, G., 2025, "AgentLesson: LLM-Based Multi-agent System for Educational Lesson Plan Generation," *International Conference on Behavioural and Social Computing*, Springer, pp. 155–170.
[52] Adam, S., 2004, "Using learning outcomes," *Report for United Kingdom Bologna Seminar*, pp. 1–2.
[53] Nicholls, D., Sweet, L., Westerway, S. C., and Gibbins, A., 2014, "The key to using a learning or skill acquisition plan," Australasian journal of ultrasound in medicine, **17**(4), pp. 141–145.
[54] Gráinne, C., 2007, "Describing learning activities. Tools and resource to guide practice," Rethinking Pedagogy for A digital Age, p. 82.
[55] Robinson Jr, J. W., 1972, "Learning Modules: A Concept," Journal of Extension.
[56] Concordia University, C. f. T. and Learning, 2021, "Teaching Academy: Resources for Promoting Teaching Excellence," https://teachingacademy.concordia.ca/courses/course-design/lessons/instructional-planning/.
[57] Ipsita, A., Erickson, L., Dong, Y., Huang, J., Bushinski, A. K., Saradhi, S., Villanueva, A. M., Peppler, K. A., Redick, T. S., and Ramani, K., 2022, "Towards modeling of virtual reality welding simulators to promote accessible and scalable training," *Proceedings of the 2022 CHI conference on human factors in computing systems*, pp. 1–21.

Appendix A: Learning Activity Library

Table 4 Learning activity library used during the cooking demonstration task.

Activity	Phase	Description
intro_pizza_types	Introduction	Introduction to different styles of pizza from around the world.
intro_dough_basics	Introduction	Introduction to pizza dough ingredients and kneading technique.
intro_sauce_cheese	Introduction	Introduction to classic pizza sauces and cheese options.
intro_baking_basics	Introduction	Introduction to baking techniques and oven safety.
presentation_pizza_tools	Presentation	Explanation of essential pizza-making tools such as rolling pins, pizza stones, and peels.
presentation_dough_steps	Presentation	Step-by-step visuals of mixing, kneading, proofing, and shaping dough.
presentation_layering	Presentation	Demonstration of the order and technique for layering sauce, cheese, and toppings.
presentation_bake_temps	Presentation	Explanation of baking temperatures and their effects on crust and toppings.
practice_tool_matching	Practice	Drag-and-drop activity matching pizza tools to their names and uses.
practice_dough_kneading	Practice	Guided kneading activity with timer and visual cues.
practice_flavor_combo	Practice	Q&A activity identifying suitable topping combinations.
practice_temp_setting	Practice	Practice simulation for setting oven temperature and baking time.
application_kitchen_setup	Application	Setup of a pizza-making workstation using appropriate tools.
application_dough_prep	Application	Preparation and resting of dough from scratch.
application_personal_pizza	Application	Creation of a personalized pizza using selected ingredients.
application_bake_and_serve	Application	Baking and serving the completed pizza.

Table 5 Learning activity library used during the welding training study.

Activity	Phase	Description
intro_mig_equipment	Introduction	Introduction to welding equipment.
intro_mig_safety	Introduction	Introduction to welding workplace safety.
intro_wire_change	Introduction	Introduction to welding wire change.
intro_tee_joint	Introduction	Introduction to Tee-joint welding.
presentation_welding_equipment	Presentation	Presentation of welding equipment commonly found in a MIG welding workplace.
presentation_mig_safety	Presentation	Video presentation on welding safety rules, PPE, and hazardous materials.
presentation_open_tension_feed_wire_close_tension	Presentation	Demonstration of feeding wire through the wire feeder.
presentation_wire_gun_set_wire_length	Presentation	Demonstration of feeding wire through the welding gun and setting the correct wire length.
presentation_open_cover_cut_wire	Presentation	Demonstration of opening the welding machine cover and cutting bent wire.
presentation_turn_on	Presentation	Demonstration of turning on the welding machine after wire feeding.
presentation_disassemble_welding_gun	Presentation	Demonstration of disassembling the welding gun.
presentation_reassemble_gun	Presentation	Demonstration of reassembling the welding gun.
presentation_avatar_demo	Presentation	Demonstration of Tee-joint welding.
practice_shielding_gas	Practice	Q&A activity identifying gas cylinder types.
practice_power_feeder	Practice	Q&A activity identifying welding machine types.
practice_flammable_liq	Practice	Removal of hazardous materials in the welding workplace.
practice_ppe	Practice	Collection of personal protective equipment in the welding workplace.
practice_wire_gun_set_wire_length	Practice	Feeding wire through the welding gun and setting the correct length.
practice_reassemble_gun	Practice	Reassembly of the welding gun.
practice_open_tension_feed_wire_close_tension	Practice	Feeding wire through the drive rolls.
practice_open_cover_cut_wire	Practice	Opening the welding machine cover and cutting excess wire.
practice_disassemble_welding_gun	Practice	Disassembly of the welding gun.
practice_turn_on	Practice	Turning on the welding machine.
practice_virtual_welding	Practice	Tee-joint welding practice activity.
application_mig_safety_equipment	Application	Application activity for welding workplace safety and equipment.
application_wire_change	Application	Application activity for welding wire change.
application_tee_joint	Application	Application activity for Tee-joint welding.

List of Figures

List of Tables