VRBCS: A Virtual Reality Application for Teaching Basic Programming Concepts to Engineering Students

Vaia Maragkou[1,*], Maria Rangoussi[1], Dimitris Metafas[1], and Errikos-Chaim Ventouras[2]

[1]Department of Electrical and Electronics Engineering, University of West Attica, Athens-Egaleo, Greece
[2]Department of Biomedical Engineering, University of West Attica, Athens-Egaleo, Greece
Email: v.maragkou@uniwa.gr (V.M.); mariar@uniwa.gr (M.R.); dmetafas@uniwa.gr (D.M.); ericvent@uniwa.gr (E.-C.V.)
[*]Corresponding author
Manuscript received October 14, 2025; revised November 17, 2025; accepted January 5, 2026; published June 16, 2026

***Abstract*—Computer programming is a highly desirable and in-demand skill; accordingly, introductory-level courses are included in engineering and science curricula. The increasing student dropout rates, however, have prompted educational research on advanced digital teaching and learning tools, such as Virtual Reality (VR), to attract and motivate students. In the present empirical study, a VR application was designed, developed in Unity 3D, and pilot-evaluated. A desktop (semi-immersive) and a head-mounted display (fully immersive) version of the same application were compared in terms of (i) educational effectiveness with novice programmers (junior engineering students), (ii) usability, and (iii) the user experience they offer. The application was effective for novice programmers; however, the difference in learning gains between the two versions was not statistically significant. In terms of usability, the fully immersive version was rated higher than the Desktop version, the difference here being statistically significant. User experience with the fully immersive version received very high ratings. Findings corroborated existing evidence on the effectiveness of VR as an educational tool for novice programmers and revealed a strong educational potential even for complex programming concepts. The findings also suggest directions for future research on the affordances of VR in education, although more comprehensive and rigorous evaluation is needed.**

***Keywords*—engineering education, computer programming, Virtual Reality (VR), immersive VR, desktop VR, Unity 3D, usability evaluation, user experience**

I. Introduction

Virtual Reality (VR) is a rapidly evolving technology that enables simulated three-dimensional user experiences and allows users to interact with virtual environments and objects therein, as if they were real [1]. The contents of VR applications, as well as the domains that benefit from such applications, have expanded to cover a broad spectrum of fields ranging from entertainment, in the form of immersive digital games [2], to medicine [3], military/defense industries [4] and education and professional training [5].

Education, in particular, is one of the areas where the advantages and affordances of VR have long been recognized and appreciated. Subjects that do not lend themselves to direct, hands-on student experience in the real world – due to safety, cost, physical dimensionality, or abstraction/complexity constraints – are now accessible via VR at an almost physical level of experience. Furthermore, recent advances in VR technologies offer enhanced user interactivity, complemented with safety and lower costs. As a result, a multitude of educational applications of VR have been and continue to be developed, to teach students in medicine [3], engineering [6], foreign languages [7, 8], history [9], chemistry, biology [10] and other sciences [11], among other disciplines. VR is also used to raise awareness and train the general public on physical disasters and emergencies, e.g., see [12] and references therein.

The technological landscape of VR is also quite varied, with existing solutions addressing either single users or many concurrent, possibly interacting users, e.g., CAVE systems [13, 14], and affording various levels of immersion and interaction. Research and development of single-user systems are today focused on (i) the 'semi-immersive' Desktop VR, where the user *views* the virtual environment on a (computer) 2D screen and *interacts* with it through common devices such as keyboard, mouse or joystick, and (ii) the 'fully immersive' or just 'immersive' Head-Mounted Display (HMD) VR, where the user *bears* a special headset or glasses with sensors that track user motion and *interacts* with the virtual environment through hand-held controllers. Desktop VR dominated the scene up to almost 10 years back, when the Oculus DK1 VR headset was launched [15, 16]; since then, HMD VR was made available to broader user groups.

Today, both Desktop VR and HMD (immersive) VR find extensive use, thanks to their complementary advantages that render each solution attractive to specific target groups or settings. In either case, the user views and interacts with a 3D application running on a computer.

- *Desktop VR* allows the users to sense their physical environment; therefore, it typically produces more modest feelings of "presence" and "immersion". However, it offers practical advantages: it relies on affordable, often already available equipment that requires minimal training; it improves safety and comfort because users do not need to move in physical space; and it supports joint viewing and possible collaboration at the same computer screen [17].
- *HMD (immersive) VR*, in contrast, insulates users from the physical environment and focuses their attention on the virtual world. The resulting increases in "presence" and "immersion" are counterbalanced by practical constraints: specialized equipment (HMD, controllers, 360° surround sound, and a high-specification computer) and user training; potential discomfort (e.g., dizziness, nausea, or headaches) that can limit session duration; the need for a larger, unobstructed physical space for safe movement; and higher development costs [18, 19].

In terms of user interaction, and regardless of the technology employed, VR applications may be grouped into two categories:

- *Experience VR*, where the user assumes a largely passive role of viewing, inspecting, watching, hearing and in

general experiencing the virtual environment, while interaction is kept to a minimum: the user may only be required to start or stop the application, or to switch to the next scene; and

- *Interaction VR*, where interaction through devices such as keyboard, mouse, joystick or hand-held controllers in HMD-based VR is emphasized and often necessary for the scenario to unfold. The user assumes a more active and participatory role, often resulting in a richer experience due to a stronger sense of presence.

Interaction VR is of great interest for education, as it facilitates the implementation of more advanced pedagogical approaches, such as active learning, discovery learning, and learning-by-doing [20]. It can therefore serve as an attractive alternative to hands-on laboratory training on subjects of human-machine interaction (driving vehicles or operating machinery) or human-human interaction (medicine, health care, social interactions). For the same reasons, Professional/Vocational Education and Training (VET) is another field that benefits from Interaction VR [21].

In either case, the reported educational and pedagogical advantages of VR are multifaceted [22]. The learners benefit from the feelings of presence and immersion created by VR technologies, in order to learn, explore and construct knowledge, possibly by trial and error, in a safe, low-cost and fear-of-failure-free framework. Immersive VR, in particular, increases learner focus, engagement and retention of the acquired knowledge, thanks to the temporary isolation from all external distractors [23, 24]. These conditions are considered ideal for developing learner skills, build learner confidence and self-efficacy, and eventually motivate the learner for lifelong learning.

In the context outlined above, the present study aims to investigate VR effectiveness for teaching basic computer programming concepts to junior engineering students. Computer programming skills are today considered highly desirable and in high demand in the job market, with relevant introductory-level courses offered in the majority of undergraduate programs of studies in engineering, sciences or other disciplines [25, 26]. However, students have been reported to drop out of these courses at increasing rates [27–30]. As a result, modern educational research has turned to exploring educational methods and tools that afford increased user interaction with the learning content, e.g., [31], and/or increased user control over the pace and path through this content, e.g., [32], in order to attract, motivate and engage students in learning computer programming. VR is a major choice, thanks to its potential for student motivation and engagement.

Along these lines, the present study employs VR technology to introduce junior Electrical and Electronics Engineering (EEE) students to basic programming concepts. This choice is motivated by two considerations: (i) traditional methods and tools (e.g., books, tables, and online whiteboards) often fail to attract and motivate students, and (ii) VR has an inherent advantage in this context because it functions both as the teaching/learning medium and as a topic taught in EEE and other relevant engineering and science disciplines.

The main challenge of this study is to assess whether a technologically advanced, attractive, and interactive medium, such as VR, leads to (i) improved learning outcomes (cognitive gains), obtained through (ii) a positive learning User eXperience (UX) (affective/emotional gains). To address these questions, an empirical, quasi-experimental study is conducted. The study is based on the design and development of a VR application Virtual Reality for Basic Coding Skills (VRBCS) whose effectiveness in evaluated in both domains, cognitive and affective, by comparing a Desktop VR version with an HMD-based immersive version.

Given the applied and empirical nature of this work, the findings are intended to inform practice rather than theory. In particular, the findings are expected to provide additional evidence on the educational effectiveness of VR and the relative merits of the different VR technologies currently available. Furthermore, the results can also inform education researchers and practitioners (including instructors/teachers and developers) in designing functional and well-accepted VR-based educational interventions. Finally, this study aims to offer insight into the potential of VR to support the teaching and learning of more advanced programming concepts in higher-education engineering and science curricula.

II. Literature Review

The use of an innovative technology such as VR for teaching and learning basic computer programming concepts and techniques has already attracted education research interest, as revealed by the set of recent publications reviewed in this section. This non-systematic review is based on Scopus, ERIC and Google Scholar and covers the years 2017–2024, when VR technology had matured enough to become affordable and accessible to education researchers and practitioners.

Among the articles retrieved, those selected should present empirical studies with users/learners that employ any type of VR technology for teaching and/or learning computer programming concepts and/or techniques; they should also include at least one evaluation activity and report results. Articles that have borderline relevance to the present research, e.g., those addressed to teachers or programmers rather than students, those proposing pedagogical models or frameworks for VR embedding, or those with purely technical orientation, are excluded. These inclusion/exclusion criteria yielded 21 publications (Table 1), which reveal a clearly increasing research interest in the field.

Table 1. Recent (2017–2024) publications of relevant empirical research

Year	Publications	Publications Count
2017	(Vincur *et al*., 2017) [33]	1
2018	(Vosinakis *et al*., 2018) [34]	1
2019	(Bouali *et al*., 2019) [35], (Tanielu *et al*., 2019) [36]	2
2020	(Jin *et al*., 2020) [37], (Segura *et al*., 2020) [38]	2
2021	(Saleh *et al*., 2021) [39], (Pirker *et al*., 2021) [40]	2
2022	(Sunday *et al*., 2022) [41], (Lu *et al*., 2022) [42], (Parmar *et al*., 2022) [43], (Alexander *et al*., 2022) [17]	4
2023	(Agbo *et al*., 2023) [44], (Gunay & Barakat, 2023) [19], (Holder *et al*., 2023) [45], (Saleh *et al*., 2023) [46]	4
2024	(Jin *et al*., 2024) [47], (Ekman *et al*., 2024) [48], (Jin *et al*., 2024) [49], (Avellar *et al*., 2024) [50], (Criollo-C *et al*., 2024) [51]	5
	Total	**21**

Common features shared among these studies are (i) the introductory level of the material that is addressed to novice programmers, and (ii) the rather young target groups, consisting mostly of junior undergraduate students.

From the technological perspective, a first finding regarding the two aforementioned technologies of Desktop VR and HMD (Immersive) VR is that despite the conspicuous recent shift towards Immersive VR, Desktop VR still attracts considerable research effort. In fact, far from being outdated, Desktop VR has increased its 'footprint' as a more accessible and readily available solution of lower cost. Indeed, 7 out of the 21 studies (1/3) involve Desktop VR implementations, while 20 out of the 21 studies involve Immersive VR, either exclusively (50%), or in comparison to Desktop VR (30%) or to traditional, non-VR methods (20%).

The use of low-cost solutions, such as Google Cardboard, is seen to attract certain research interest, especially in studies that address larger user groups, to suppress cost per capita [35, 39, 44, 42]. Because these low-cost solutions are not equipped with motion sensors, user discomfort is frequently reported (dizziness, nausea, headaches) due to the misalignment between user movements in the physical and the virtual spaces [41, 46]. It does not come as a surprise, therefore, that 'smartphone VR' popularity is declining, e.g., [44].

From the educational perspective, the majority of the reviewed studies include evaluation/measurement of (i) *user experience / satisfaction*, as perceived after having used the app, (18 out of the 21 studies, or 85%), (ii) *usability* of the employed VR app, as measured, e.g., through the System Usability Scale (SUS) Questionnaire or other similar tools (13 out of the 21 studies, or 62%), and (iii) *user attitudes* towards / acceptance of a novel technology such as VR being introduced in education (16 out of 21 studies, or 76%). Despite the educational orientation of these studies, only 11 out of them (practically 50%) include evaluation and report evaluation results on student learning gains/benefits.

Among the studies that do measure (positive) learning gains are [34] on Logic Programming on component-based programming (Scratch-like) [36, 44, 43] on computational thinking, and [46] on Python programming. The last one reports a statistically significant difference between learning gains of the experimental group (VR) and the control group (traditional teaching). On the contrary, the differences in performance between experimental and control groups measured by Lu *et al*. [42] on student spatial visualization skills, by Jin *et al*. [49] on teaching Science, Technology, Engineering, Art and Mathematics (STEAM) and by Gunay and Barakat [19] on teaching graphical programming, are not found to be statistically significant. Positive results as to the user-perceived learning gains based on self-reporting rather than objective measurements are reported by Alexander *et al*. [17, 47], while increased learning gains of the experimental versus the control group are objectively measured in [43, 44].

Another finding has to do with the lack of a common basis regarding the methodology, protocols and measurement tools for the *evaluation* of educational VR applications. Consequently, the results obtained should be interpreted with care, given that researchers have to resort to custom evaluation procedures and tools which in turn produce outcomes prone to a certain degree of subjectivity and therefore not directly comparable.

Regarding the level of programming skills sought, the most popular choices are visual or block-based or component-based programming applications, along the tradition of Scratch. These choices are compatible with the target groups of novice programmers; furthermore, they indicate a policy of avoiding the more demanding form of coding in a typical programming language such as C, C++, Java or Python, in favor of more intuitive, self-explanatory and 'cold-start' forms that would allow the learner to concentrate on programming concepts rather than the 'technicalities' of a language [38, 45, 47, 49, 50]. Of interest to the present study is that more challenging programming concepts, such as *nested loops* and *recursion*, are not included.

Considering the technological and educational perspectives together, it may be claimed that educational VR environments and applications have not yet fully exploited the potential of VR technology. In particular, studies are inclined towards more elementary and simple representations that can be adequately portrayed by 2D technologies, blackboards, whiteboards, paper sheets with code written on them, panels and other types of flat surfaces being typical such examples. This trend may be ascribed to the intention of the researchers to keep the cognitive load low and to avoid burdening/distracting the learners—VR users by more advanced and fancy 3D elements technically affordable in VR.

III. The Present Research: Motivation, Research Questions and Novelty

Motivated by the studies reviewed in the previous section, and by the strengths and weaknesses identified in the educational uses of VR, the present research aims to propose, design, develop, and evaluate a novel educational VR application, namely, Virtual Reality for Basic Coding Skills (VRBCS), to teach basic coding concepts and techniques to novice programmers (here, junior EEE students). Furthermore, Desktop VR and Immersive VR technologies are comparatively evaluated as to their educational effectiveness (learning gains of the users/learners), usability, and user experience. Specifically, the Research Questions addressed in the present study are:

1) What is the *educational effectiveness* (learning gains of users/learners) of the immersive VRBCS version in teaching basic programming concepts and techniques to novice programmers, as compared to the Desktop version with the same content?
2) What is the *usability* of the immersive VRBCS version, as compared to the Desktop version with the same content?
3) What is the *user experience* of the immersive VRBCS version, as compared to the Desktop version with the same content?

The pedagogical rationale for the development of VRBCS, like any other VR-based educational tool/technology, goes back to theories of learning that emphasize the role of experience and active participation. Specifically, the theories of learning by doing, active learning, discovery learning, and experiential learning, as proposed by Dewey [52] and later by Lewin [53] and Kolb [54, 55], posit that immediate

experience obtained via active participation (e.g., as in a hands-on laboratory), and mediated experience obtained via reflection and analysis following active participation, are beneficial for learners, as they enhance learning outcomes. Recall and retention of new knowledge, in particular, are found to be enhanced, thanks to the strong mark that experience leaves on learners' minds as compared to indirect knowledge obtained through reading or studying a subject. Similar (positive) results on learning gains are reported by research studies that employ VR technologies [56–59]. The educational effectiveness of VR technology draws exactly from experience and active participation: it is ascribed to the potential of VR to provide students with "almost real" experiences and to facilitate experimentation—a great advantage regarding domains of knowledge that do not allow direct experience because of cost, safety, dimensionality or abstraction / complexity issues. Computer programming falls under the latter class: the abstract and complex nature of this particular subject challenges and discourages students. VR technology emerges as an ideal solution for the development of educational scenarios that address the knowledge and skills mastery along with the motivation and engagement targets at once. VRBCS, in particular, aspires to build on these VR advantages in order to offer students concrete knowledge, yet, via a more attractive and motivating path.

Is Research Question 1 on the educational effectiveness of VRBCS trivial or not worth investigating? Studies that report neutral or even negative findings in terms of learners' knowledge gains via VR, such as [60] or [61], make further research necessary.

The learning content of VRBCS is defined so as to cover basic programming concepts at increasing levels of difficulty:

1) simple (not nested) loops, to be repeated a set number of times,
2) calling of simple (non-recursive) functions and parameter passing,
3) nested loops, for 2D data structures and problems,
4) branching based on logic conditions (if-then-else type), and
5) recursion—calling recursive functions.

The target group is junior EEE undergraduate students, who are novices in computer programming and not familiar with VR technologies (e.g., gamers are excluded). The two versions of VRBCS, Immersive and Desktop, are to be comparatively evaluated on the same learning content and education scenario.

The major specifications for the design and development of VRBCS are as follows:

- an interactive character that provides the user with immediate visual feedback as to the consequences of any action, decision or choice made,
- a gamified educational character, yet, with minimal game elements (not a full game, not DGBL),
- a non-competitive character, that is based on the use of bonuses but not penalties,
- a not time-pressured character, without timeouts, to avoid stressing the user.

This specific research targeting and framework certainly have common elements with existing research regarding the learning content, the target group, and the interactive and gamified character, while also being innovative in certain other aspects. One such aspect is the comparison of the Immersive and Desktop VR technologies, which is performed here on exactly the same learning content and scenario, in contrast with existing studies that proceed to compare these technologies on different content and on similar but not identical scenarios [19, 48]. Such use of a common learning basis is considered critical to isolate the effect of the VR technology from other interfering factors.

A second innovative element of the current research is the evaluation tools employed. Since the two versions of the VR application essentially differ only (i) as to the level of immersion and (ii) as to the means of user interaction (keyboard/mouse versus hand-held controllers), the evaluation questionnaires users will be asked to complete are also designed to be essentially identical as to (i) the learning gains (knowledge test), and (ii) usability of the application, in order to guarantee comparability of the results. A third, separate questionnaire is employed for the evaluation of the immersive VR version on user experience; this is justified by the distinct character of the immersive VR version. This setup is decided so as not to decrease the comparability of the results of the first two tests/questionnaires.

Another novel element of this study is the strict filtering of the volunteering students to select the final experimental group, in order to ensure homogeneity of the group (junior EEE students, novices in programming, no VR experience/not gamers).

Last but certainly not least is the novelty in the learning content covered by the present study: nested loops and recursive function calling are topics not addressed in existing studies, because of their more complex level. The pedagogical approach of increasing level of difficulty adopted by the current study, however, is consistent with the inclusion of demanding content, such as recursion, as a final stage / VR scene of the content.

IV. Design and Development of VRBCS

VRBCS comprises eight scenes with progressively more demanding learning content, from the first (Scene-1) to the last one (Scene-8). The Unity 3D engine is used to develop the same 8 scenes in both the Desktop and Immersive versions of VRBCS. For the development of the immersive version, the HTC Vive Pro headset is employed, equipped with movement sensors and hand-held controllers. Table 2 provides an overview of the five programming concepts taught and the respective VRBCS scenes that elaborate on each concept.

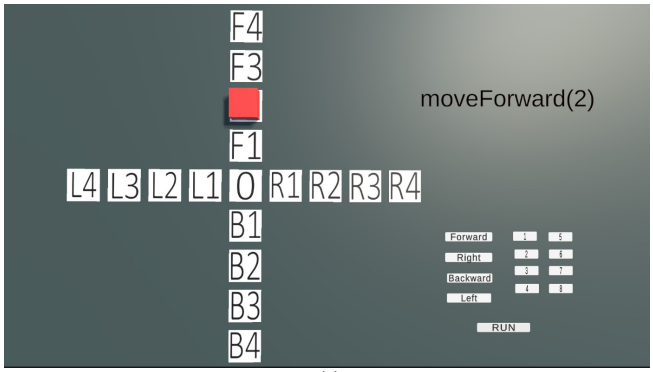
Table 2. Programming concepts and relevant scenes in VRBCS

Concept	VRBCS Scenes
(i) simple (not nested) loops, to be repeated a set number of times	Scene-1, Scene-2, Scene-3, Scene-4
(ii) calling of simple (non-recursive) functions and parameter passing	Scene-1, Scene-2, Scene-3, Scene-4
(iii) nested loops, for 2D data structures and problems	Scene-5
(iv) branching based on logic conditions (if-then-else type)	Scene-6, Scene-7
(v) recursion—calling recursive functions	Scene-8

A. Simple Loops and Calling of Simple Functions

These two concepts are jointly introduced and practiced in

the first four scenes, Scene-1 to Scene-4. In either of these scenes, the user is prompted to select the appropriate buttons in order to move an object (the rotation-invariant red cube or the rotation-variant smiley face) to a target position. The transport is completed in a number of identical one-step movements, repeated for a number of times; the number is calculated and inserted by the user. The scenario is implemented as a simple *for-loop* included in a simple function that is called when the user presses the RUN button for the scenario to start unfolding.



(a)

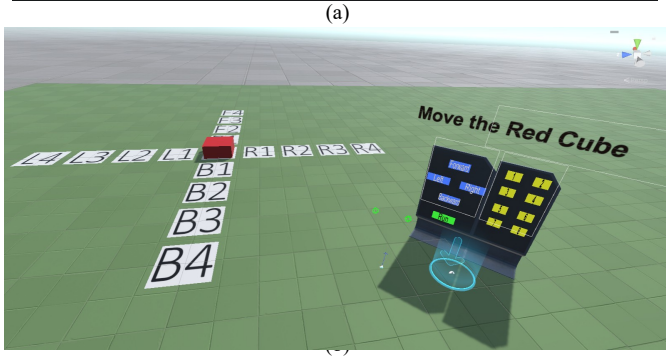
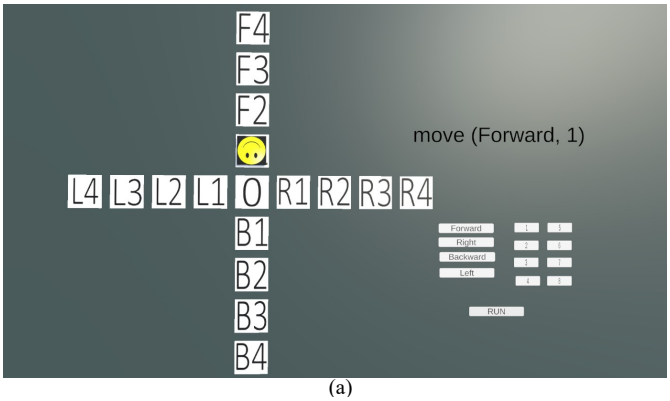
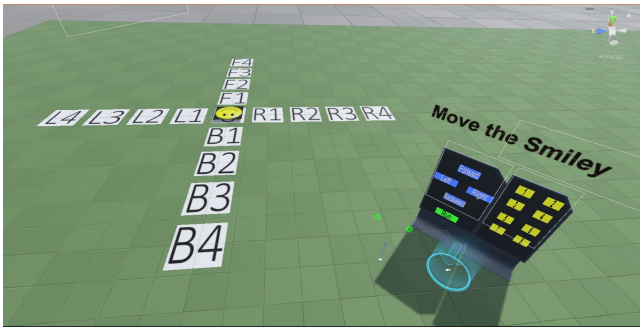


Fig. 1. Simple loops and simple function calling with 1 parameter. (a) VRBCS-desktop version; (b) VRBCS-immersive version.



(a)



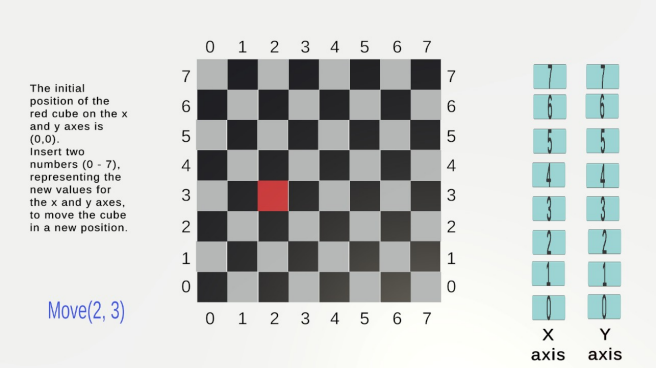
(b)

Fig. 2. *Simple loops* and *simple function calling* with 2 parameters. (a) VRBCS-desktop version; (b) VRBCS-immersive version.

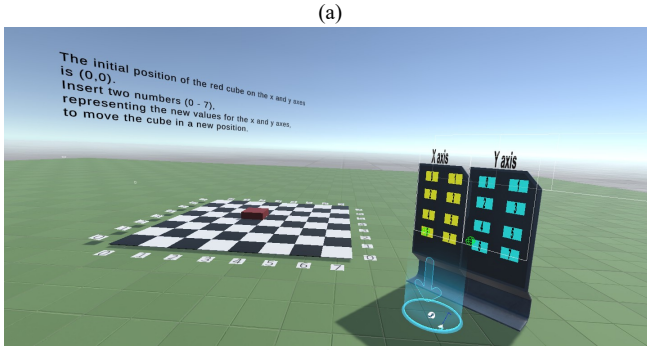
All four scenes involve movement along a single axis (1D problems) although multiple 1D axes may be available. The user may experiment with either calling one of four direction-specific functions, namely, *moveForward, moveBackward, moveLeft, moveRight* and passing a single parameter (number of iterations—steps along that direction), e.g., Fig. 1, or calling the unique, direction-independent function *move*, and passing two parameters (direction and number of iterations—steps), e.g. Fig. 2.

B. Nested Loops for 2D Data Structures and Problems

Random movement on the 2D plane is selected as a suitable analogy to introduce the students to 2D data structures and problems. In particular, a familiar 8x8 chessboard is selected to depict the 2D plane (Fig. 3). The user is prompted to move an object (a red cube) to a selected destination cell, by setting the values of X and Y coordinates of the destination. 'Building' on knowledge from preceding scenes, a single *move* function is called with two parameters (X and Y coordinates of the destination cell); here, however, the implementation of the *move* function includes two *for-loops*, with one nested inside the other. The outer *for-loop* proceeds along X direction. For each X value of the outer *for-loop*, the inner *for-loop* moves the object along the Y direction, until the destination cell is reached.



(a)



(b)

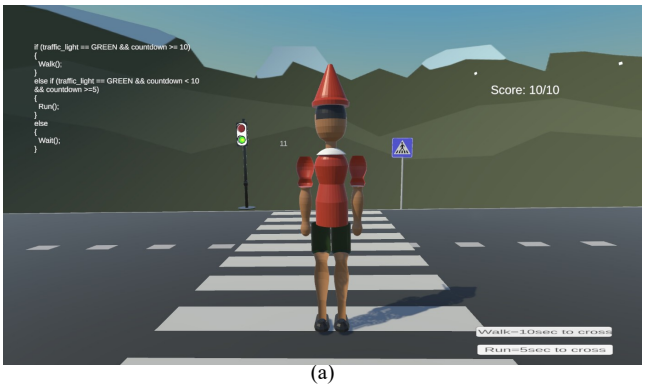
Fig. 3. *Nested for-loops* in a single *move* function, to move an object on the 2D plane. (a) VRBCS-desktop version; (b) VRBCS-immersive version.

C. Branching Based on Logic Conditions (If-Then-Else)

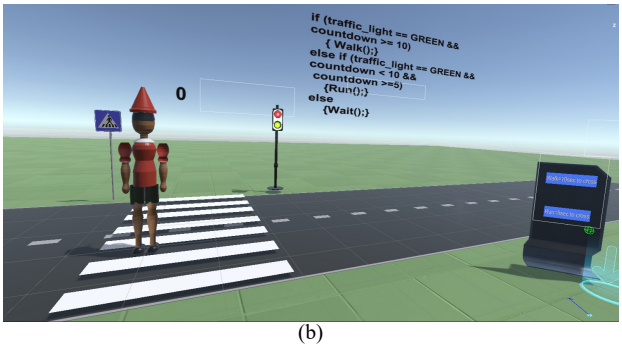
The concept of program branching based on logic conditions is introduced through the familiar situation of a pedestrian that has to decide when and how to cross a street safely. The crossing is equipped with a timer, and safe crossing can be achieved by walking or running, when the light is green. A human-like figure (Pinocchio) is used to enhance users' feeling of presence. A points bonus is introduced as a gamification element (Fig. 4). The user should first read and understand the conditional statements shown in pseudo-code:

```
if (traffic_light == GREEN && countdown >=
      10)
  Walk();
else
  if (traffic_light == GREEN && countdown <
      10 && countdown >=5)
    Run();
  else
    Wait();
```

It is a nested if-then-else type of conditional code that implements a *tri-furcation* branching logic. Scenes with multiple successive pedestrian crossings are developed for training.



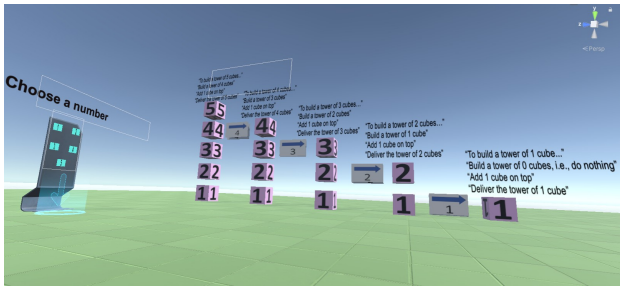
(a)



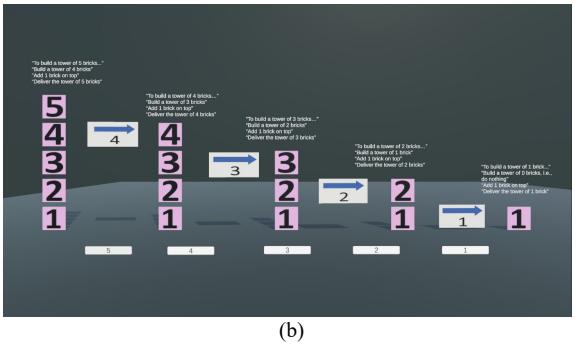
(b)

Fig. 4. Tri-furcation *branching based on logic conditions* (if-then-else type). (A) VRBCS-desktop version; (B) VRBCS-immersive version.

D. Recursion—Calling of Recursive Functions



(a)



(b)

Fig. 5. *Recursion*—calling of a recursive function to build a tower of 5 blocks. (a) VRBCS-desktop version; (b) VRBCS-immersive version.

Recursion is an essential concept in computer programming that students often find challenging to understand. The paradigm selected to introduce this concept is building a tower of N blocks, as it is both simple and intuitive (Fig. 5). The scenario unfolds (i) in a top-down, problem-decomposition direction, where building a tower of N blocks is broken down to an identical but 'smaller' problem of building a tower of N-1 blocks, plus an 'easy' step of adding 1 block on top, and then (ii) in a bottom-up, solution-composition direction, where N blocks are successively added to build the tower and solve the original problem. It is implemented as a recursive function called with the number of blocks (N) as a parameter. Both scenarios are explained by appropriate pseudo-code on screen.

E. Discussion on Design and Development

The structure and design of VRBCS is dictated by a set of practical considerations, constraints and education-related decisions discussed in this section.

The extent of VRBCS (5 concepts, 8 scenes) is set after consideration of the requirements of the learning content covered, jointly with the maximum time the user could comfortably remain in the Immersive VR version, which is generally considered to be limited to a continuous session of 30 min. This 'outer' time frame is not imposed internally to the scenes, however. In the scenes, timers and time-outs are avoided to allow users take their time in each scene and to avoid increasing their stress while they will be striving to address at once the demands of the equipment and the learning content. In fact, it is the learning content per se that dictates this 'loose' time management, since it is expected that the time each user needs to observe and study each scene in detail and respond to the tasks in it may vary significantly, depending on individual background. Another concern as to the Immersive VR version is that a user might feel like removing the HMD and taking a short break due to discomfort, dizziness or headache; a scene should not time out in such cases, in order to allow the user to get back and seamlessly continue when feeling comfortable. Consequently, no limit is set as to the time the user remains in a certain scene, the number of repetitions of the same scene or the overall time the user spends in VRBCS.

Graphical elements are intentionally kept simple, in order to emphasize interaction and usability over decoration and to avoid distracting users from the learning goals. In the Immersive VR version, the look and feel of the virtual environment are uniform across all scenes; they all feature an open, exterior space. The user stands on a 2D plane (the ground) of green color that extends to the horizon; blue skies are used above the horizon. Besides their intuitive nature that makes the user feel all is normal and therefore safe in the virtual world, these choices are meant to also give the user a feeling of ample space without physical limitations. A similar, yet, more simple type of environment is adapted for the Desktop version.

In the scenes where the user has to proceed through selection of values, the selection mechanism is implemented as virtual buttons, in both versions. In the Desktop version, the user has to click upon these buttons using the mouse or the keyboard (typical choice in computer games), while in the Immersive version the buttons are mounted on one (or two, where necessary) selection panels and the user selects the

proper one by pointing at it with the hand-held controllers. For the user's convenience, button categories are color-coded depending on functionality, with the same color code used in both versions.

The educational character is dominant in both versions; gamification elements are embedded in certain scenes to render the application more attractive, yet, not in the form of a complete digital game that would justify the characterization as Digital Game-Based Learning (DGBL). The two versions are designed and developed to be educationally fully equivalent, their major differentiation being the type of user interaction, as mentioned above.

Since users are selected so as not to be familiar with the use of VR, they need briefing, directions, and some time to familiarize themselves with the virtual environment, the equipment (HMD, controllers), the way to move around (teleporting) and the way to handle virtual objects through controllers. In order to accommodate this need, that is more evident with immersive version users, the learning content of the first scene is intentionally kept to a minimum, in order to have it serve as an introductory step.

V. Pilot Evaluation and Results

A. Evaluation Methodology, Sample and Tools

The pilot evaluation of VRBCS was conducted by volunteering junior undergraduate EEE students of the University of West Attica (UNIWA), Athens-Egaleo, Greece, who enrolled in the 1st semester of studies in academic year 2024-25.

The research protocol submitted to the UNIWA Research Ethics Committee for approval included a mandatory informed consent form signed by all volunteers. The form clearly stated the research aims and objectives, procedures, tools and measures against potential hazards (e.g., discomfort in Immersive VR). Students' rights were also clearly stated, including (i) participation on a purely voluntary basis, (ii) the right to leave the project at any time and without any academic or other consequence, (iii) the protection of their personal data as per GDPR, through confidential handling by the research group and anonymized presentation of the results of the study, (iv) the storage of only the absolutely minimum data necessary for the objectives of the study, where no sensitive personal data are to be included, (v) the publication of aggregate, statistical analysis results and no identifiable individual cases, and (vi) the secure storage of collected raw personal data according to the data protection legislation in effect (GDPR or other).

Upon approval of the research protocol, a call for volunteers was addressed to all students enrolled in the EEE.1.5 'Introduction to Programming' 1st year undergraduate course. A grade bonus in the EEE.1.5 course laboratory (mandatory unit) was offered for full participation in this study. The students could choose to evaluate the Desktop or the immersive version of VRBCS, or they could accept being placed in either of the two conditions by the researchers. Two experimental groups were thus formed: Group A (19 students) for the Desktop version and Group B (14 students) for the immersive version. Homogeneity within each group and equivalence of the two groups as to (i) background knowledge on programming and (ii) experience in using VR, were ensured thanks to the strict admission criteria employed.

Methodologically, the study is a quasi-experiment [62] using convenience sampling, two experimental conditions (Desktop VR vs. Immersive VR), and no control group. The latter choice was made because at this initial stage the aim is not to compare VR educational effectiveness to that of conventional teaching. This type of evaluation would require a careful alignment of the learning content to the specific chapters taught in the hosting course per week, and a synchronization of the VRBCS evaluation process with the class teaching along the academic semester. Following such practical considerations, evaluation of VRBCS with a control group is planned as a next phase of this study.

The evaluation tools employed comprise 1 knowledge test and 2 different questionnaires, completed online by the students. These are:

1) A knowledge test, developed by the authors and tailored to the learning content covered by VRBCS. It consists of 15 closed-ended questions, 13 of the multiple-choice type and 2 of the matching type.
2) A usability questionnaire developed by the authors as an extended version of the standard System Usability Scale (SUS), developed by John Brooke in 1986. SUS has gained popularity and has become a standard tool for usability evaluation of any online environment [63]. It consists of 10 questions in the form of statements aimed at assessing user experience of such a system as to its usability. Users have to rate their agreement with each statement on the 5-point Likert scale, from 1 = 'fully disagree' to 5 = 'fully agree'. To better serve the purposes of the current study, here SUS is extended into eSUS, by appending 7 more items to the original 10, to obtain a set of 17 items, 14 of which are statements requiring agreement scores on the 5-point Likert scale (quantitative analysis), while the last 3 items are open-ended questions requiring free text answers (qualitative analysis).
3) A custom User eXperience (UX) questionnaire. This is based on a similar concept questionnaire proposed and used by [64], adapted in order to accommodate the immersive version of VRBCS. It consists of 10 statements to be answered on the 5-point Likert scale from 1 to 5 and it is aimed at assessing the experience of a user that has used and interacted with an Immersive VR application.

The knowledge test and extended usability questionnaire are completed by Group A and Group B students, in order to obtain comparable results as to the educational efficiency and the usability of VRBCS, while the user experience questionnaire is completed only by Group B students.

The knowledge test, in particular, was administered as a post-test only, because the strict admission criteria employed, along with the student status (1st year students in their 1st semester of study), rendered a pre-test on programming knowledge redundant and possibly discouraging for the volunteers.

Regarding evaluation procedure and settings,

- The 19 students in Group A were asked to download the executable file of the Desktop version of VRBCS from UNIWA Open E-class platform to their computer and run it asynchronously at home. To complete the task and gain

the bonus grade, they should go through all eight scenes and then complete the knowledge test and extended usability questionnaire described in the previous paragraph. A loose time limit of a week was set, so as not to pressure the students but to ensure that both items would be completed soon after running VRBCS.

- The 14 students in Group B were asked to run the immersive version of VRBCS in the Electronics and Computers Laboratory of the aforementioned Department of EEE in UNIWA Campus II, Building Z, Lab ZB-110. This room is equipped with HTC Vive headsets, computers with adequate resources to run Unity 3D and the immersive version of VRBCS, as well as a free space of adequate surface and equipped with sensors adjusted for Immersive VR testing. A timetable was arranged according to individual student availabilities. Students were admitted in the Lab in groups of 3. The first author briefed each group on VR technology and its educational uses, on the aims and objectives of the present study and on the VR equipment they would use to test VRBCs. The next step was actual VRBCS testing; this was performed individually by each user and lasted for approximately 20 minutes, student-dependent. The first author was standing by, in the same room, to make sure the tests would run smoothly and the users would feel safe and comfortable. Upon completion, the student would be asked to complete all 3 evaluation items described in the previous paragraph, while the next student would proceed to use the HMD and controllers and test the application.

B. Knowledge Test Results

Knowledge test was completed online by 33 users in total, 19 students in Group A, anonymized here as A_ST_01 to A_ST_19 and 14 students in Group B, anonymized here as B_ST_01 to B_ST_14. The actual test was not anonymous, because of the bonus grade offered to reward volunteering students for participation. The 15 questions in the test were graded on the 0 to 10 scale; maximum student score was thus 150. Personal average grade and standard deviation across the 15 questions of the test were calculated for each student and linearly scaled to the 0–100 range, for easier interpretation. Results in the form of personal average grades ± 1 standard deviation are shown in Fig. 6(a) and Fig. 6(b), per student in Group A [B], while Group-average grades, normalized to the 0–100 scale, are shown in Fig. 7 per question.

The average grade in Group A was 86.89 over 100.00 with a standard deviation of 33.95; individual student grades ranged between 63 and 94 (Fig. 6a). In comparison, average grade in Group B was 87.79 over 100.00 with a standard deviation of 32.43; individual total scores ranged between 78 and 100 (Fig. 6b). These results are sufficiently close to indicate that there is no significant difference between the two versions of VRBCS as to student knowledge and learning gains. A *t-test* is performed to test this (null) hypothesis of zero difference of the means in the two Groups, A and B. The two groups are not paired (independent samples) and the degrees of freedom are 31. The 2-tailed t-test value is $t = -0.3527$ ($p = 0.7266$, for $a = 0.05$). The null hypothesis cannot be rejected, i.e., the difference in the means of the two Groups is not statistically significant.

Fig. 7 provides a closer look at the contribution of each question in the knowledge test to student knowledge. The first 2 questions are more challenging to the students, regardless of group they belong, as only 2 or 3 correct answers are given in each question. A possible explanation is that these 2 questions focus on content not explicitly presented in VRBCS; rather, the student is expected to observe the details in the respective scene in order to fully understand the concept. In contrast, the majority of students in either group give correct answers to all other questions, despite their progressively more demanding content.

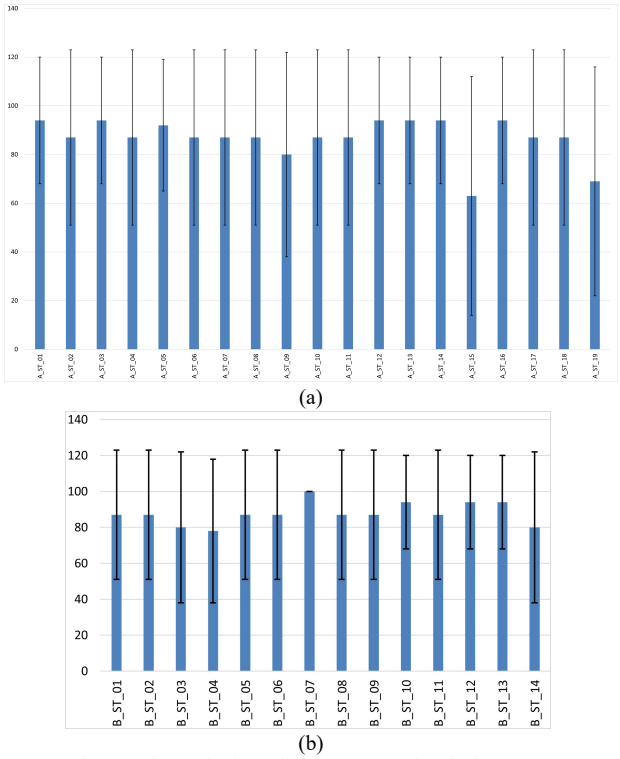


Fig. 6. Student grades in the knowledge test completed after trying VRBCS. Horizontal axis: anonymized student ID. Vertical axis: personal grade normalized to 0 – 100 (average ± 1 SD). (a) Group A students. (b) Group B students.

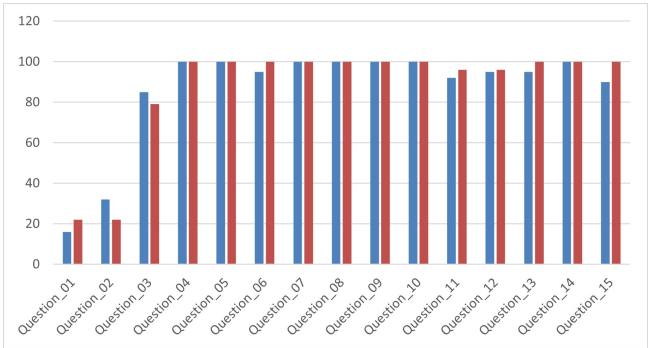


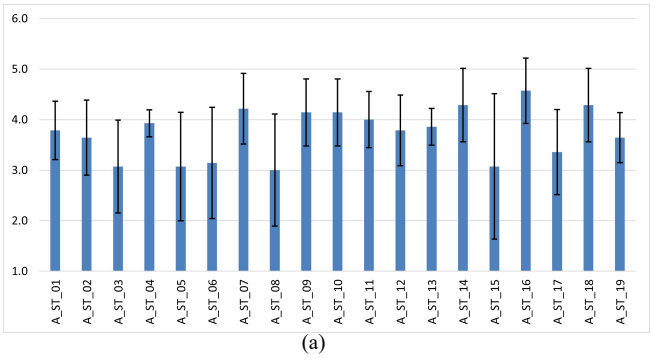
Fig. 7. Comparison of Group A and Group B average grades across the 15 questions of the knowledge test. Blue: Group A. Red: Group B.

C. Extended System Usability Scale Questionnaire Results—Quantitative Part

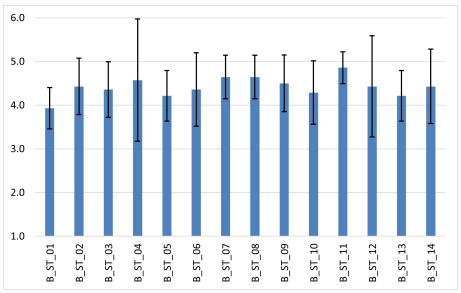
The results obtained by analyzing student answers to the extended System Usability Scale (eSUS) questionnaire are presented here separately for items 1–14 (quantitative analysis), and for items 15–17 (qualitative analysis). The reliability of the quantitative part of eSUS is first checked by Cronbach's α [65], which was $\alpha = 0.859$ (ideally, $0.70 < \alpha <$

1.0), and by the average inter-item correlation of the 14 items (questions), which was $r = 0.313$ (ideally, $0.20 < r < 0.40$). Both values indicate that eSUS is internally consistent, i.e., all 14 items consistently measure the same underlying concepts / dependent variables. Given that the original SUS is reported to have Cronbach's α at the level of 0.90 to 0.92, the value obtained here indicates that the modification adopted in the present study has not severely affected its internal consistency. Furthermore, it was found that Cronbach's α does not increase when any one of these items is left out of the eSUS questionnaire, meaning that none of the 14 questions is redundant.

The eSUS results, in the form of personal average scores ±1 standard deviation, are shown in Fig. 8(a) and Fig. 8(b) per student in Group A [B], while Group-average scores are shown in Fig. 9 per question. Given that the original SUS questionnaire arranges its 10 questions in alternating positive-negative direction (wording)—a pattern repeated also in the eSUS questionnaire—before proceeding with the quantitative analysis, responses to negatively worded items were reverse-coded to 5–1 instead of 1–5, to align all items in the same direction and obtain meaningful statistical results.



(a)



(b)

Fig. 8. Student scores in the extended usability test completed after trying VRBCS. Horizontal axis: anonymized student ID. Vertical axis: personal score, on the 5-point Likert scale (1 – 5), average ± 1 SD. (a) Group A. (b) Group B.

In Fig. 8(a), Group A average score is 3.63 ±0.49 (average ±1 standard deviation); it conveys a moderately positive student view on the usability of the Desktop version. Personal average scores range from 4.57 (A_ST_16) to 3.00 (A_ST_08) over 5.00; extreme values differ by only 1.57 over 5.00, showing moderate convergence and homogeneity among Group A student views. Standard deviations range from 1.43 (A_ST_15) to 0.27 (A_ST_04) over 5.00; the difference of 1.17 over 5.00 is also indicative of a moderate level of agreement among Group A students.

In Fig. 8(b), Group B average score is 4.41 ±0.22 (average ±1 standard deviation); it conveys a more positive student view on the usability of the immersive version as compared to Group A student view regarding the Desktop version. Personal average scores range from 4.96 (B_ST_11) to 3.93 (B_ST_01) over 5.00; extreme values differ by 0.93 over 5.00, showing an increased level of convergence and homogeneity among Group B student views. Standard deviations range from 1.40 (B_ST_04) to 0.36 (B_ST_11) over 5.00; the difference of 1.04 over 5.00 is also slightly lower than the respective Group A value of 1.17, indicating a higher level of agreement among Group B students.

The overall picture portrayed by Fig. 8 is that Group B has increased average scores accompanied by lower standard deviations, as compared to Group A. It can be claimed, accordingly, that although both versions were evaluated on the positive side, usability of the immersive version was more positively regarded than that of the Desktop version. This is verified by the *t-test* performed to test the null hypothesis of zero difference of the means in the two Groups, taken in the sense 'Group A – Group B'. The two groups are not paired (independent samples) and the degrees of freedom are 27. The 2-tailed t-test value is $t = -5.311$ ($p = 1.320 \cdot 10^{(-5)}$, for $a = 0.05$). The null hypothesis is rejected, i.e., the difference in the means of the two Groups is statistically significant.

The results in Fig. 8 are presented per question in Fig. 9. Group A [Group B] average ±1 standard deviation is 3.74 ±0.26 [4.40 ±0.29]. Furthermore, group-averaged scores per question are higher in Group B than in Group A for each and every question. Consequently, Group B students find the immersive version of increased usability compared to Group A students and their evaluation of the Desktop version. This result corroborates findings in Fig. 8 above.

A closer inspection of Group A (Desktop version) answers in Fig. 9 shows that Question_01 ('*I think I would like to use this system frequently*') is the least positively regarded one. Indeed, it received a moderate 3.05 over 5.00, indicating that possibly students did not find this version attractive and/or interesting enough, despite its ease of use and the asynchronous, 'your-time-your-place' protocol used with them. On the contrary, Question_02 ('*I found this system unnecessarily complex*') is the most positively (originally, least negatively) regarded one, with a 4.05 over 5.00 in the reversed logic (1.95 over 5.00 in the original logic), indicating that students found the system straightforward to use.

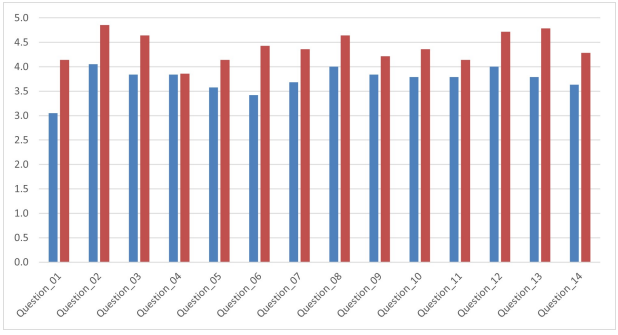


Fig. 9. Comparison of Group A and Group B average student answers to the eSUS questionnaire (items 1–14), on the 5-point Likert scale (1–5). Blue: Group A. Red: Group B.

A similar type of inspection of Group B answers shows that the Question_04 ('*I think I will need assistance by a technical person in order to be able to use this system*') received a moderate 3.6 over 5.00, indicating that students did feel the need for a certain degree of technical support

while using the immersive VRBCS version. This finding verifies that it was meaningful both to brief Group B students before using the equipment and to have one of the researchers (first author) present in the room and standing by while the students used the immersive version of VRBCS in the lab.

D. Extended System Usability Scale Questionnaire Results—Qualitative Part

The last 3 questions in the eSUS questionnaire are open-ended and require free-text answers. Although they do not exactly refer to usability, they are appended to eSUS because they are deemed necessary for a comprehensive evaluation of the two VRBCS versions.

Question_15 'What was the best element of this application / the one you liked the best?'

- Group A: The majority of students singled out Pinocchio and the scenes where he appeared (Scene-6 and Scene-7) as the best part of the application (11 users), followed by Smiley who rotated to face the correct direction before moving along it, in Scene-2, Scene-3 and Scene-4 (4 users). In more detail, elements students referred to as their favorite were interactivity, the fact that Pinocchio would move on their 'command' (pressing the walk or the run button), the fact that the code was made part of a game and held the role of instructions, as well as the gamification elements and the chance to gain points. Several users made positive comments on the application as being 'well programmed' and 'smoothly running'.
- Group B: The simple, easy-to-use character of the application along with the clear directions and the immediate feedback were the top elements (4 users), followed by Pinocchio and the scenes where he appeared and where the user had to read and understand the code in order to act properly and get the points (3 users). Immediate feedback was also praised in these scenes. The increasingly demanding content along the 8 scenes was referred to by 3 users and the interaction opportunity the user was given in the virtual world as well as the smooth moving in the virtual space through teleporting were also referred to by 3 more users. An overall positive experience was mentioned by almost all 14 students who stated that they found the application attractive and motivating as well as educative through a direct and experiential learning method.

Question_16 'What was the worst element of this application / the one you liked the least?'

- Group A: The majority of students replied that they considered the application as more or less successful, although 16 out of the 19 students did refer to points that they did not like or thought that should be re-examined. Scene-1 could be omitted as rather trivial (5 users); Scene-8 should better explain its content (the concept of recursion) (4 users); the sound of cars horns as Pinocchio was run over in Scene-6 and Scene-7 was unpleasant (1 user); all scenes would benefit from a smoother way to exit, e.g., an exit button (a few users); all scenes would benefit from more detailed directions in case of user error (a few users).
- Group B: Out of the 14 users, 6 did not identify any negative point; rather, they described their experience as clearly positive. The other 8 students raised the following points which they characterized as 'minor comments': Scene-1 was rather trivial; it could be made more interesting by allowing the user to move more freely in the virtual space; Scene-8 (recursion) was not interactive enough; the arrangement of selection buttons on panels could be improved; certain technical issues should be addressed.

Question_17 'What changes would you suggest for further developing this application in the future?

- Group A: Although all students expressed positive overall opinions on the application tested, the majority of them had interesting ideas to propose for the improvement and extension of the application. Major elements referred to were better graphics and richer visual elements—in the Pinocchio scenes, for example, cars should be shown running on the street, or Pinocchio should flex his legs when walking or running. Increased interactivity of the user was proposed; more detailed introduction and directions as well as feedback signs to the user as to his/her progression along the material were deemed necessary throughout the scenes. The moving objects should be allowed to make more free / complex movements; a minor application bug detected should be corrected; a smoother way to exit a scene should be provided.
- Group B: More complex and demanding scenes and tasks were proposed by 4 users; certain of the eight scenes could merge to result in a smaller, yet more challenging set; the user should be given more choices and increased autonomy; increased user interactivity should be introduced; more scenes; better graphics. Last but not least one of the students notes: 'I would like it to extend into more scenes and further on to extend to other subjects taught in our curriculum. It is very useful to enter a virtual world and to be able to see how things one designs or studies in theory would look like or function in practice'.

E. User Experience Questionnaire Results

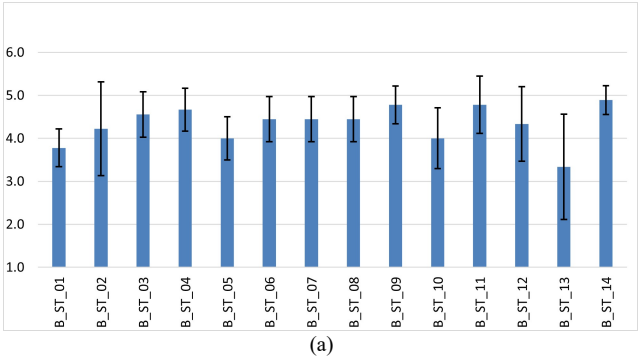
The User Experience (UX) questionnaire, consisting of 10 questions-statements to be answered on the 5-point Likert scale (1–5), was first checked with the Cronbach's α, calculated as $\alpha = 0.626$ (ideally, $0.70 < \alpha < 1.00$), and with the average inter-item correlation of the 10 items (questions), calculated as $r = 0.187$ (ideally, $0.20 < r < 0.40$). Both values are lower than anticipated, indicating an internal consistency problem. It was found, however, that both statistics increased when one particular question, namely, Question_04 "*I did not feel uncomfortable/unconfident whenever the buttons were out of my field of view*", was removed from the set of 10; furthermore, removal of any other single question did not have an increasing impact on these two statistics. A closer look in the results revealed that Question_04 combined low average score with high variability (3.93 ±1.14 over 5.00). Therefore, this question was removed and the UX finally used consisted of 9 out of the original 10 questions. The modified UX questionnaire of 9 items was validated before reuse, on the same data used to validate the original questionnaire. Cronbach's α was recalculated as $\alpha = 0.758$ and average inter-item correlation was recalculated as $r = 0.267$. To further validate internal consistency of the modified UX questionnaire, the MacDonald's omega (ω) coefficient was calculated [66]. This is considered as a stronger indicator of internal consistency than Cronbach's α and under more loose assumptions. The value obtained was ω

= 0.825, which indicated strong internal consistency (ideally, $0.8 < \omega < 1.00$). Therefore, all three statistics yielded acceptable values indicating that the modified UX questionnaire is internally consistent, while none of the 9 questions is redundant.

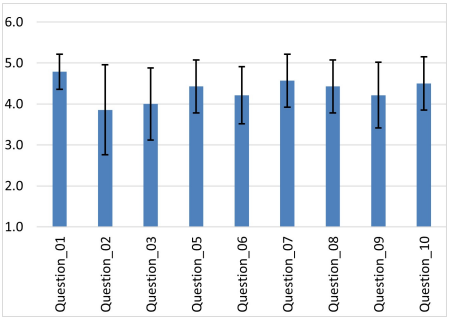
The modified UX was completed by the 14 students in Group B (immersive version of VRBCS), immediately after having used the application. Results are shown per student in Fig. 10(a) (personal average score ±1 standard deviation), and per question in Fig. 10(b) (group-averaged score for each question ±1 standard deviation). All 9 items (questions-statements) are positive-directed, their interpretation therefore is that the most favorable answers are towards 5.

Results in Fig. 10(a) reveal a high average—low deviation score of 4.33 ± 0.43, with the majority of personal average scores ranking above 4.00 over 5.00 and the majority of standard deviations calculated below 1.00—a result that indicates strongly positive user experiences with the immersive VRBCS version, along with clear views and confident answers by the majority of the students.

Fig. 10(b) shows that all 9 questions are favorably answered by the students, as average values range from 3.86 to 4.79 over 5.00, with a total average of 4.33 ± 0.29. In fact, Question_01 ('*I thought this VR application was easy to use*') is the one answered to the most positive, followed by Question_07 ('*I did not need much time to select the combinations of buttons for my answer*'), while Question_02 ('*I thought there was not much inconsistency in this application*') is the one answered to the least positive. Students have therefore found the immersive version of VRBCS chiefly easy to use and of intuitive, self-explanatory design, although they express certain concerns about its consistency.



(a)



(b)

Fig. 10. Group B student answers to the UX questionnaire, on VRBCS immersive version. (A) Personal average scores per student. Horizontal axis: anonymized student ID (1-14). Vertical axis: Personal average score across the 10 questions (average ± 1 SD). (B) Group-averaged scores per question. Horizontal axis: Question number (1-9). Vertical axis: Group-averaged score per question (average ± 1 SD).

VI. Discussion

A. Answers to the Research Questions

Research Question 1: What is the educational effectiveness (learning gains of users/learners) of the immersive VRBCS version in teaching basic programming concepts and techniques to novice programmers, as compared to the Desktop version with the same content?

Based on the results in section V.B, it may be claimed that both versions of VRBCS are effective as tools for teaching and learning basic programming concepts and techniques. The high scores obtained by students in either Group (approximately 87/100), as shown in Figs. 6 and 7, are an objective measure of this effectiveness. Furthermore, the differences in student performance between the two groups are not pronounced enough to establish superiority of the Immersive or the Desktop version; the two versions are in fact found to be practically equivalent in that aspect.

This outcome is in good agreement with existing studies that report practically equivalent effectiveness of Immersive and Desktop VR in the cognitive domain [19, 47, 49], although certain other studies report superior educational efficiency of the Immersive VR [43]. Such disagreement may be due to the fact that evaluation methods, protocols, sample and tools have not yet been standardized; the results produced are therefore not directly comparable. In a broader perspective, the same outcome is in good agreement with all existing studies that report generally positive learning gains through VR [17, 36, 44–46].

The learning equivalence between Immersive and Desktop VR has significant implications for the cost and scalability of VR technology in education. Despite higher user evaluations, Immersive VR requires specialized and costly equipment (HMD, sensors, dedicated space and powerful computers). On the other hand, Desktop VR is far more easily scalable, because space and equipment already available in any computer laboratory can be utilized. It is therefore an advantageous solution for low budget and/or large student cohort conditions. In contrast, the number of available devices limits the number of participants in Immersive VR. A promising solution is to combine Immersive VR for small groups or more specialized and demanding learning subjects/scenarios and Desktop VR for general integration into the curriculum, in order to achieve both pedagogical effectiveness and sustainable use of resources.

Current VR technology offers low-cost and easy-to-use immersive VR systems, such as smartphone holders with glasses, of significantly reduced interaction and user experience quality, as well as advanced VR systems that do not require sensors and/or external computer(s) and offer a good quality-price balance. Given the variety of existing VR solutions, the choice should result from jointly analyzing technical and financial terms against the needs of the specific use case.

Research Question 2: What is the usability of the immersive VRBCS version, as compared to the Desktop version with the same content?

Both VRBCS versions received positive evaluations as to their usability, with an average score of 3.63 (±0.49) for the Desktop version (Fig. 8a) as compared to 4.41 (±0.22) for the immersive version (Fig. 8b). These findings are in agreement

with existing studies that also report positive user ratings on the usability of various VR applications, e.g. [17, 33, 36, 39, 41, 45, 47, 49, 51].

Usability of the immersive version was more positively perceived and commented on by students in comparison to usability of the Desktop version, as established by the results in Sections V.C and V.D (Figs. 8 and 9). Although this may seem counterintuitive, the same result is reported by [48]. It may possibly be ascribed to the fact that all students are practically familiar with desktop applications and tend to have higher expectations from any application running on this technology. On the other hand, given the strict selection filter applied in the selection of participants, Group B students who tried the immersive VRBCS version were not familiar with Immersive VR, neither as programmers nor as gamers. The feelings of presence and immersion generated in them may have resulted in a greater enthusiasm with this new technology that has led them ignore possible usability issues.

A distinct component is the usability of the Immersive VR technology per se, regardless of the specific educational application that employs it. Users may complain about the complexity of bearing the HMD and holding and using the hand-held controllers, e.g., [39, 48]. In addition, Immersive VR is often reported to produce different levels and types of discomfort to certain users, such as headaches, dizziness, nausea or imbalance, the impact being stronger after longer HMD sessions, e.g. [46]. This aspect of immersive VR (user discomfort) has not been explicitly evaluated in the present study; none of the Group B students did complain of any type of discomfort, however, either when completing the extended usability questionnaire or the user experience questionnaire.

Research Question 3: What is the user experience of the immersive VRBCS version, as compared to the Desktop version with the same content?

On the basis of results in section V.E (Fig. 10), the experience with the immersive VRBCS version was perceived as strongly positive by Group B students. Indeed, students gave an enthusiastic average score of 4.33 (±0.43) over 5.00. This finding is cross-verified with student comments in the free-text answers of the last 3 questions in the usability questionnaire (section V.D) that essentially assess user experience. Although the immersive VR version is technically more demanding to develop and is therefore expected to generate more modest user experiences, the highly positive student responses are probably accounted for by the novelty of experiencing a new technological tool. Similar positive user experience is reported in existing studies [17, 19, 33, 36, 42, 45–51].

A final comment is that usability of any given digital application constitutes only one of the factors that decide user experience, the latter being a much broader concept. Although this distinction is recognized and in fact reflected in the different evaluation tools employed in certain studies, e.g., [39, 46, 48, 49], in certain other studies the two terms are used as near-synonyms and users are asked to evaluate both concepts in a single questionnaire or survey of mixed questions, e.g., [36, 38, 43, 41, 45] – a practice that may be confusing for certain users. In the latter cases, the results reported (positive in the vast majority of the cases) cannot safely be credited either to usability or user experience.

In view of the above discussion, it may be claimed that the findings of the present study have contributed to the open research question on the educational effectiveness of VR technologies mainly regarding the comparison between the immersive and the semi-immersive type.

- Findings on the educational practical equivalence of these two types (Research Question 1) add to the evidence base of similar evidence reported by existing studies and balance against reports on the opposite that keep emerging.
- Positive findings on student experience with Immersive VR (Research Question 3) corroborate existing reports on similar user attitudes under practically all tested contexts—all begging for longitudinal studies to isolate the novelty effect, however.
- On the other hand, the interest of findings based on the eSUS usability questionnaire (Research Question 2) is internal to the current application (VRBCS): they are valuable in order to inform future development of this application and may prove useful to the wider research community to the degree that future VRBCS versions prove successful.

A final point worth stressing is that findings support the strong potential of VR in teaching and learning not only elementary but also more advanced core programming concepts, such as nested loops and recursive functions. This finding certainly calls for a closer investigation in future research, since this potential of VR has not been brought out in existing studies which often focus on younger ages and/or employ suitable block-based / Scratch-like design and tools.

Novel elements of this research are (i) the comparison of Immersive and Desktop VR versions of VRBCS on exactly the same learning content and scenario, (ii) the use of common evaluation tools to assess student gains in the cognitive domain (knowledge tests) and the usability of VRBCS (eSUS questionnaire), for both versions, (iii) the strict filtering of the volunteering students (junior EEE students, novices in programming, no VR experience/not gamers), in order to ensure homogeneity of the group, and (iv) the learning content that covers nested loops and recursive function calling—two more advanced programming concepts that are not covered in existing similar research studies.

Major aspects of interest regarding potential wider adoption and usage of VRBCS are flexibility, modular structure, simple and straightforward user interface and a gradually increasing level of difficulty suitable for novice programmers. Thanks to these features, VRBCS may be adapted for use across various target groups / learner ages and may accommodate different requirements as to technology and cost. The Desktop version, in particular, is a cost-effective solution suitable for embedding in typical engineering / sciences curricula.

B. Limitations of this Research

Limitations that may have affected the results of the current study are methodological and practical, the latter including equipment issues, volunteering student issues, and timing issues.

Methodologically, the self-reported nature of eSUS and UX questionnaires is a potential threat to the validity of the results. Indeed, self-reporting may introduce bias, as

evaluators may tend to give socially desirable answers, may misunderstand or misinterpret questions, or may not remember facts accurately. Anonymization, triangulation and provision of example answers, along with the use of mixed methods and validated scales are employed to address this issue. In the present study, validated scales have not been identified for use and anonymization was applied only in reporting and publishing the results. Internally, student IDs are known to the researchers, as students have to log on to the university e-class platform with institutional credentials and work there, in order to get the grade bonus offered for participation. However, triangulation is indeed employed as approximately 1/3 of the questions in eSUS are also present in UX in a rephrased form, and respective results are cross-verified. This approach is legitimate since usability (measured by eSUS) is a decisive factor for user experience (measured by UX). Mixed methods are also employed in eSUS, where the 3 open-ended questions (items 15–17) qualitatively analyzed in Section V.D are in fact used to cross-verify findings in the quantitative part of the same tool (items 1–14).

Another concern regarding the validity of the results, also of methodological origin, comes from the limited sample size, especially that of Group B (Immersive VR condition). Students did respond to the call for volunteers, but not in the numbers expected—an understandable behavior, given that the trials would coincide with the beginning of their first semester of study, a rather pressing and demanding period for them. Sample size was further reduced by the strict admission filter applied, which resulted in a percentage of the initial number of volunteers eventually being admitted to the study.

Evaluation of student learning gains on the basis of only a post-test without a pre-test, according to the rationale given in Section V, is another decision that may have had impact on the results obtained, especially those referring to the cognitive domain.

Certain other conditions that may have influenced the results of this study refer to the volunteer recruitment and Group formation method. In particular, students were allowed to select experimental condition (Group A or Group B) and a different testing environment was used per condition (home for Group A versus lab for Group B). Although these decisions were dictated by practical considerations as to the feasibility of the study and the need to secure a minimum number of volunteers, yet, it is possible that they have affected the validity of the results obtained—for example, they may have led to a sample with a higher interest or motivation for new technologies, which may have reinforced the positive attitude towards VR. The different settings for each group may have affected the concentration, anxiety or overall experience of the participants. Another potential threat comes from the well-known *novelty effect* associated with innovative technologies such as Immersive VR. An Immersive VR application may generate excitement and result in more positive evaluations due to the increased senses of presence and immersion it causes to the user. In the present study, students assigned to the immersive condition were required not to have prior experience with immersive VR, a factor that may have amplified initial enthusiasm and produced the *novelty effect*, potentially inflating UX ratings and impressions of usability. Moreover, a stronger sense of presence and immersion may lead users to overlook minor control difficulties or inconsistencies that might be judged more strictly in a more familiar environment (Desktop VR). Therefore, the high scores in the UX questionnaire may suffer a positive bias due to the novelty effect. It is worth noting, however, that an opposite bias factor is also possibly present in the same results: Immersive VR users had to quickly get acquainted with several new interaction and control elements, which may cause fatigue (cognitive load) or confusion—especially when users must apply them after only a brief demonstration.

These methodological limitations mean that the results should be interpreted as indications rather than definitive proof. At this early stage of the VRBCS application, such results certainly retain their value, as they offer insights and inform future design / redesign decisions. A more extensive evaluation of the two VRBCS versions, however, based on a larger and—ideally—a randomized sample, a more rigorous protocol in controlled, uniform settings and a time / repetition planning to overcome the VR novelty effect, is necessary in order to obtain statistically significant and generalizable results. Indeed, in order to validate the trends observed in the present pilot evaluation through more robust statistical inference, our future work will focus on:

1) larger-scale deployments: given that an introductory computer programming course is included in the undergraduate curricula of all the 7 departments in the Faculty of Engineering of our institution, 1st year undergraduate students may become users/evaluators of VRBCS, on the basis of an inter-departmental research design; and
2) longitudinal study designs: repeated sessions and follow-up measurements (e.g., delayed post-tests) will be employed to examine the stability of learning outcomes and user experience over time, assess knowledge retention, and reduce potential biases stemming from short-term exposure effects (e.g., novelty).

Practical limitations stem from space, equipment and time constraints. The laboratory used for the Immersive version evaluation trials has only one empty room free for VR testing, as well as a single non-production system (computer, HMD, sensors, controllers) available to be used by students. Although evaluation time schedule was based on student personal availabilities, the fact that students had to queue up for the trials prolonged the overall time needed to complete evaluation; the process would certainly benefit from multiple systems and a second suitable empty room for trials.

Another limitation having to do both with the admission filter and with timing consideration is the decision to have only novice programmers evaluate VRBCS. The planning of the evaluation tests had therefore to align to the semester teaching plan of the relevant undergraduate course, Introduction to Programming, so as to place the tests early enough in the semester, when the students would have not been introduced to the programming concepts included in the VRBCS learning content.

A final comment has to do with the admission filter limitation and the intentional exclusion of students who are gamers and/or have experienced (immersive) VR. Despite the methodological benefit of this decision, i.e., the extrication of the independent factor of technology

familiarity from the evaluation of the 3 dependent factors selected (learning outcomes, usability, user experience), it is possible that the decision had an impact on the results. Immersive VR, for example, might have received a more positive or a more negative evaluation, if users were already familiar with VR technology and therefore positively or negatively predisposed towards it, or held higher or lower expectations from it. Such an impact is reported in [38], for example, where expert users rate the usability of the respective immersive VR application lower than new users.

VII. Conclusion

VRBCS, a novel application for teaching and learning basic programming concepts and techniques was designed, developed, and evaluated in a pilot study. It is addressed to junior EEE students who are novice programmers and is available in two versions, a Desktop version and an Immersive version. Pilot evaluation assessed (i) learning outcomes, through a knowledge post-test, (ii) usability of the application, through the eSUS questionnaire, and (iii) user experience, through a custom UX questionnaire. Although users responded much more positively to the immersive version, and also rated this version higher in terms of usability, difference in the learning outcomes achieved by the two versions was not found to be statistically significant.

According to the evaluation results, future research should focus on (i) improving the design of the VRBCS environment, to make gamification elements more pronounced, to improve graphics and aesthetics, and to increase user interaction with the virtual objects, (ii) increasing user autonomy and control of the application and scene sequencing, and (iii) providing more detailed introduction and help to the user. As an example, student comments on Scene-1 contents being elementary or on Scene-8 not being interactive enough, provide valuable feedback for the design / redesign of future application versions. Scene-1 may be replaced by an introductory scene aimed solely at user familiarization, without pure learning content on the subject taught. User interaction in Scene-8, on the other hand, can be enhanced through the introduction of gaming elements in it. Evaluation results will thus be immediately utilized to improve both the learning experience and the level of interaction.

Although the evaluation performed was only a pilot, quantitative as well as qualitative analysis of the results indicate significant educational potential of both the Desktop and the immersive version of VR for teaching and learning computer programming to novices. A more comprehensive evaluation with a larger sample, a control group and a more rigorous protocol is necessary in order to establish the validity of the results. Another methodological improvement may stem from the adoption of a pre-test on student background knowledge on the subject of computer programming. Inclusion of a pre-test in future work will strengthen the conclusions because the pairwise comparison of student performance will allow a more accurate estimation of VRBCS impact on student knowledge and therefore stronger claims about students' incremental cognitive progress (learning gains). In addition, we plan to administer a delayed post-test at a time interval after the use of the application, in order to measure knowledge retention and how it compares with student performance immediately after using VRBCS.

Two questions that have emerged during the current study and that certainly merit a more careful look in the next steps of this study are (a) whether the high levels of acceptance by the students and positive attitude towards (Immersive) VR would be sustained in the case of formal introduction and systematic use of this technology in the educational process, or the enthusiasm, interest and engagement would fade out with prolonged use (the *novelty effect*), and (b) how could (immersive) VR technology be leveraged to create and deploy advanced pedagogical and educational scenarios of discovery learning, constructivist learning and even collaborative learning, given that the majority of use cases, including VRBCS, have had a more or less behaviorist orientation up to now.

Besides these directions, future research could also focus on the emotional aspects of learning through VR, examining how immersion affects emotional engagement and motivation and their contribution to learning outcomes. In addition to immediate learning outcomes, another question worth researching is whether the degree of immersion affects long-term retention and memorization. Results would shed light on how VR technology affects not only the cognitive dimension but also the emotional and temporal components of the learning experience. Finally, in the same context, elements worth investigating are those related to the personalization of both the learning content and user experience in general.

As a final note, immersive VR applications have already proved successful in educating, training, or raising awareness of various target groups or the general public on diverse topics beyond typical education, such as learning foreign languages, training the general public for resilience to disasters/emergencies, etc. In light of this observation, and despite the limitations of this study discussed in the previous section. VRBCS is a modular educational platform that is flexible enough to hold promise for adaptation and extensive use either within or beyond the scope of computer programming.

Research Ethics

This study was approved by the Research Ethics Committee (R.E.C.) of the University of West Attica, Greece (Approval: Act 32nd/29-11-2024, Decision number 120598 / 13-12-2024).

Informed consent was obtained from all individual participants included in the study.

Conflict of Interest

The authors declare no conflict of interest.

Author Contributions

Conceptualization: all authors; Methodology: VM, MR, DM; Design and Development: VM; Supervision: MR, DM, E-CV; Writing: VM; All authors approved the final text.

Funding

The authors declare that no financial support was received for the research of this article. The APC was funded by the

Special Account for Research Grants (SARG) of the University of West Attica, Athens-Egaleo, Greece, as per UNIWA SARG Act 28/06-09-2022.

ACKNOWLEDGMENT

Authors wish to thank the undergraduate students of the Department of Electrical and Electronics Engineering of UNIWA who volunteered for this study.

REFERENCES

[1] J. Bailenson, *Experience on Demand: What Virtual Reality Is, How It Works, and What It Can Do*, New York, NY, USA: W. W. Norton & Company, 2018.

[2] C. Cruz-Neira, M. Fernández, and C. Portalés, "Virtual reality and games," *Multimodal Technologies and Interaction*, vol. 2, no. 1, 8, 2018. doi: 10.3390/mti2010008

[3] M. Javaid and A. Haleem, "Virtual reality applications toward medical field," *Clinical Epidemiology and Global Health*, vol. 8, no. 2, pp. 600–605, 2019. doi: 10.1016/j.cegh.2019.12.010

[4] C. Armas, R. Tori, and A. V. Netto, "Use of virtual reality simulators for training programs in the areas of security and defense: A systematic review," *Multimed. Tools Appl.*, vol. 79, no. 5–6, pp. 3495–3515, Feb. 2020. doi: 10.1007/s11042-019-08141-8

[5] A. A. Mazhar and M. M. Al Rifaee, "A systematic review of the use of virtual reality in education," in *Proc. 2023 Int. Conf. Inf. Technol. (ICIT)*, 2023, pp. 422–427. doi: 10.1109/ICIT58056.2023.10225794

[6] M. Soliman, A. Pesyridis, D. Dalaymani-Zad, M. Gronfula, and M. Kourmpetis, "The application of virtual reality in engineering education," *Appl. Sci.*, vol. 11, no. 6, 2879, 2021. doi: 10.3390/app11062879

[7] V. Maragkou and M. Rangoussi, "French as a foreign language: Design, development and evaluation of an immersive virtual reality learning environment," in *Proc. Int. Technol., Educ. Develop. Conf. (INTED)*, Valencia, Spain, 2018. doi: 10.21125/inted.2018.2345

[8] R. D. Pinto, B. Peixoto, M. Melo, L. Cabral, and M. Bessa, "Foreign language learning gamification using virtual reality—A systematic review of empirical research," *Educ. Sci.*, vol. 11, no. 5, 222, 2021. doi: 10.3390/educsci11050222

[9] J. Parong and R. E. Mayer, "Learning about history in immersive virtual reality: Does immersion facilitate learning?" *Educ. Technol. Res. Dev.*, vol. 69, no. 3, pp. 1433–1451, 2021. doi: 10.1007/s11423-021-09999-y

[10] T.-F. Chuang, Y.-H. Chou, J.-Y. Pai, C.-N. Huang, H. Bair, A. Pai, and N.-C. Yu, "Using virtual reality technology in biology education," *Amer. Biol. Teach.*, vol. 85, no. 1, pp. 23–32, Jan. 2023. doi: 10.1525/abt.2023.85.1.23

[11] P. Šiđanin, J. Plavšić, I. Arsenić, and M. Krmar, "Virtual Reality (VR) simulation of a nuclear physics laboratory exercise," *Eur. J. Phys.*, vol. 41, no. 6, 065802, 2020. doi: 10.1088/1361-6404/ab9c90

[12] V. Maragkou, M. Rangoussi, I. Kalogeras, and N. S. Melis, "Educational seismology through an immersive virtual reality game: Design, development and pilot evaluation of user experience," *Educ. Sci.*, vol. 13, no. 11, 1088, 2023. doi: 10.3390/educsci13111088

[13] C. Cruz-Neira, D. J. Sandin, T. A. DeFanti, R. V. Kenyon, and J. C. Hart, "The CAVE: Audio visual experience automatic virtual environment," *Commun. ACM*, vol. 35, no. 6, pp. 64–72, Jun. 1992. doi: 10.1145/129888.129892

[14] P. Marayong, P. Shankar, J. Wei, H. Nguyen, T. Z. Strybel, and V. Battiste, "Urban air mobility system testbed using CAVE virtual reality environment," in *Proc. IEEE Aerosp. Conf.*, Big Sky, MT, USA, 2020, pp. 1–7. doi: 10.1109/AERO47225.2020.9172534

[15] P. Rubin, "The inside story of oculus rift and how virtual reality became reality," *WIRED*, May 2014.

[16] P. Snow, "Virtual reality and pain management: The need for clarity for future interventions," *Brit. J. Pain,* vol. 19, 2, pp. 68–70, Apr. 2025. doi: 10.1177/20494637251319500

[17] B. Alexander, Y. Hou, B. Khan, and J. Jin, "Learn programming in virtual reality? A case study of computer science students," in *Proc. IEEE Global Eng. Educ. Conf. (EDUCON)*, 2022. doi: 10.1109/EDUCON52537.2022.9766621

[18] V. Biener, S. Kalamkar, N. Nouri, E. Ofek, M. Pahud, J. J. Dudley *et al.*, "Quantifying the effects of working in VR for one week," *IEEE Trans. Vis. Comput. Graph.*, vol. 28, no. 11, pp. 3810–3820, Nov. 2022. doi: 10.1109/TVCG.2022.3203103

[19] C. Gunay and R. Barakat, "Immersive gamification for education: No additional benefit gained from wearing a VR headset," in *Proc. 24th Annu. Conf. Inf. Technol. Educ. (SIGITE '23)*, New York, NY, USA: ACM, 2023, pp. 136–141. doi: 10.1145/3585059.3611423

[20] J.-C. Chen, Y. Huang, K.-Y. Lin, Y. Chang, H.-C. Lin, C.-Y. Lin, and H.-S. Hsiao, "Developing a hands-on activity using virtual reality to help students learn by doing," *J. Comput. Assist. Learn.*, vol. 36, no. 1, pp. 46–60, Feb. 2020. doi: 10.1111/jcal.12389

[21] D. Kablitz, M. Conrad, and S. Schumann, "Immersive VR-based instruction in vocational schools: Effects on domain-specific knowledge and wellbeing of retail trainees," *Empir. Res. Vocat. Educ. Train.*, vol. 15, no. 1, 9, 2023. doi: 10.1186/s40461-023-00148-8

[22] J. Radianti, T. A. Majchrzak, J. Fromm, and I. Wohlgenannt, "A systematic review of immersive virtual reality applications for higher education: Design elements, lessons learned, and research agenda," *Comput. Educ.*, vol. 147, 103778, 2020. doi: 10.1016/j.compedu.2019.103778

[23] C. Ochs and A. Sonderegger, "The interplay between presence and learning," *Front. Virtual Real.*, vol. 3, 742509, 2022. doi: 10.3389/frvir.2022.742509

[24] T. F. Lonne, H. R. Karlsen, E. Langvik, and I. Saksvik-Lehouillier, "The effect of immersion on sense of presence and affect when experiencing an educational scenario in virtual reality: A randomized controlled study," *Heliyon*, vol. 9, no. 6, e17196, Jun. 2023. doi: 10.1016/j.heliyon.2023.e17196

[25] I. Zhurakovskaia, J. Vézien, C. Hosson, and P. Bourdot, "Immersive serious games for learning physics concepts: The case of density," in *Virtual Reality and Mixed Reality*, P. Bourdot, M. Alcañiz Raya, P. Figueroa, V. Interrante, T. W. Kuhlen, and D. Reiners, Eds., ser. *Lecture Notes in Computer Science*, Cham, Switzerland: Springer, vol. 13105, 2021, pp. 156–170. doi: 10.1007/978-3-030-90739-6_12

[26] A. Gordillo, D. López-Fernández, and J. Mayor, "Examining and comparing the effectiveness of virtual reality serious games and LEGO serious play for learning Scrum," *Appl. Sci.*, vol. 14, no. 2, 830, 2024. doi: 10.3390/app14020830

[27] A. Robins, J. Rountree, and N. Rountree, "Learning and teaching programming: A review and discussion," *Comput. Sci. Educ.*, vol. 13, no. 2, pp. 137–172, 2003. doi: 10.1076/CSED.13.2.137.14200

[28] T. Beaubouef and J. Mason, "Why the high attrition rate for computer science students: Some thoughts and observations," *ACM SIGCSE Bull.*, vol. 37, no. 2, pp. 103–106, Jun. 2005. doi: 10.1145/1083431.1083474

[29] S. Bergin and R. Reilly, "The influence of motivation and comfort level on learning to program," in *Proc. 17th Workshop Psychol. Program. Interest Group (PPIG '05)*, Brighton, UK, 2005, pp. 293–304.

[30] J. Bennedsen and M. E. Caspersen, "Failure rates in introductory programming," *ACM SIGCSE Bull.*, vol. 39, no. 2, pp. 32–36, Jun. 2007. doi: 10.1145/1272848.1272879

[31] H.-T. Wu, P.-C. Hsu, C.-Y. Lee, H.-J. Wang, and C.-K. Sun, "The impact of supplementary hands-on practice on learning in introductory computer science course for freshmen," *Comput. Educ.*, vol. 70, pp. 1–8, Jan. 2014. doi: 10.1016/j.compedu.2013.08.002

[32] R. Takács, J. T. Kárász, S. Takács, Z. Horváth, and A. Oláh, "Successful steps in higher education to stop computer science students from attrition," *Interchange*, vol. 53, pp. 637–652, 2022. doi: 10.1007/s10780-022-09476-2

[33] J. Vincur, M. Konopka, J. Tvarozek, M. Hoang, and P. Navrat, "Cubely: Virtual reality block-based programming environment," in *Proc. 23rd ACM Symp. Virtual Real. Softw. Technol. (VRST '17)*, New York, NY, USA: ACM, 2017, 84, pp. 1–2. doi: 10.1145/3139131.3141785

[34] S. Vosinakis, G. Anastassakis, and P. Koutsabasis, "Teaching and learning logic programming in virtual worlds using interactive microworld representations," *Brit. J. Educ. Technol.*, vol. 49, no. 1, pp. 30–44, Jan. 2018. doi: 10.1111/bjet.12531

[35] N. Bouali, E. Nygren, S. S. Oyelere, J. Suhonen, and V. Cavalli-Sforza, "Imikode: A VR game to introduce OOP concepts," in *Proc. 19th Koli Calling Int. Conf. Comput. Educ. Res. (Koli Calling '19)*, New York, NY, USA: ACM, 2019, 27, pp. 1–2. doi: 10.1145/3364510.3366149

[36] T. Tanielu, R. 'Akau'ola, E. Varoy, and N. Giacaman, "Combining analogies and virtual reality for active and visual object-oriented programming," in *Proc. ACM Conf. Global Comput. Educ. (CompEd '19)*, New York, NY, USA: ACM, 2019, pp. 92–98. doi: 10.1145/3300115.3309513

[37] Q. Jin, Y. Liu, Y. Yuan, L. Yarosh, and E. S. Rosenberg, "VWorld: An immersive VR system for learning programming," in *Proc. 2020 ACM Interact. Des. Children Conf.: Ext. Abstr. (IDC '20)*, New York, NY, USA: ACM, 2020, pp. 235–240. doi: 10.1145/3397617.3397843

[38] R. J. Segura, F. J. del Pino, C. J. Ogáyar, and A. Rueda, "VR-OCKS: A virtual reality game for learning the basic concepts of programming,"

Comput. Appl. Eng. Educ., vol. 28, no. 1, pp. 31–41, Jan. 2020. doi: 10.1002/cae.22172

[39] A. Y. Saleh, G. S. Chin, R. Tei, M. K. Othman, F. S. Mohamad, and C. J. Chen, "Learning scope of Python coding using immersive virtual reality," in *Innovative Systems for Intelligent Health Informatics*, F. Saeed, F. Mohammed, and A. Al-Nahari, Eds., ser. *Lecture Notes on Data Engineering and Communications Technologies*, Cham, Switzerland: Springer, vol. 72, pp. 1029–1038, 2021. doi: 10.1007/978-3-030-70713-2_97

[40] J. Pirker, J. Kopf, A. Kainz, A. Dengel, and B. Buchbauer, "The potential of virtual reality for computer science education—Engaging students through immersive visualizations," in *Proc. IEEE Conf. Virtual Real. 3D User Interfaces Abstr. Workshops (VRW)*, Lisbon, Portugal, 2021, pp. 297–302. doi: 10.1109/VRW52623.2021.00060

[41] K. Sunday, S. S. Oyelere, F. J. Agbo, M. B. Aliyu, O. S. Balogun, and N. Bouali, "Usability evaluation of Imikode virtual reality game to facilitate learning of object-oriented programming," *Technol. Knowl. Learn.*, vol. 28, pp. 1871–1902, 2023. doi: 10.1007/s10758-022-09634-6

[42] J. Lu, L. Seavey, S. Zuk, J. Dimino, and F. Martin, "Evaluating student spatial skills learning in a virtual reality programming environment," in *Proc. 2022 IEEE Frontiers Educ. Conf. (FIE)*, 2022, pp. 1–9. doi: 10.1109/FIE56618.2022.9962403

[43] D. Parmar, L. Lin, N. Dsouza, S. Joerg, A. Leonard, S. B. Daily, and S. V. Babu, "How immersion and self-avatars in VR affect learning programming and computational thinking in middle school education," *IEEE Trans. Vis. Comput. Graph.*, vol. 29, no. 8, pp. 3698–3713, Aug. 2023. doi: 10.1109/TVCG.2022.3169426

[44] F. J. Agbo, S. S. Oyelere, J. Suhonen, and M. Tukiainen, "Design, development, and evaluation of a virtual reality game-based application to support computational thinking," *Educ. Technol. Res. Dev.*, vol. 71, pp. 505–537, 2023. doi: 10.1007/s11423-022-10161-5

[45] R. Holder, M. Carey, P. Walder, and P. Keir, "MoonBase VR: Learning to program in a virtual reality game," in *Proc. 8th Int. Conf. Inf. Educ. Innov. (ICIEI '23)*, New York, NY, USA: ACM, 2023, pp. 74–80. doi: 10.1145/3594441.3594454

[46] A. Y. Saleh, G. S. Chin, M. K. Othman, F. S. Mohamad, and C. J. Chen, "Immersive visualization of Python coding using virtual reality," *Int. J. Adv. Sci. Eng. Inf. Technol.*, vol. 13, no. 1, pp. 336–347, 2023. doi: 10.18517/ijaseit.13.1.16028

[47] Z. Jin, Y. Bai, W. Song, Q. Yu, X. Yue, and X. Jia, "DVRT: Design and evaluation of a virtual reality drone programming teaching system," *Comput. Graph.*, vol. 125, 104114, 2024. doi: 10.1016/j.cag.2024.104114

[48] J. Ekman, J. Solsona, and L. Quintero, "Codeseum: Learning introductory programming concepts through virtual reality puzzles," in *Proc. ACM Int. Conf. Interact. Media Exp. (IMX)*, 2024, pp. 192–200. doi: 10.1145/3639701.3656306

[49] Z. Jin, Y. Bai, W. Song, Q. Yu, and X. Yue, "EduCodeVR: VR for programming teaching through simulated farm and traffic," *Vis. Comput.*, vol. 41, pp. 4931–4955, 2025. doi: 10.1007/s00371-024-03699-3

[50] G. Martins Nunes Avellar, M. L. Fioravanti, W. Simao de Deus, K. R. L. J. Castelo Branco, and E. F. Barbosa, "SSPOT-VR: An immersive and affordable mobile application for supporting K-12 students in learning programming concepts," *Educ. Inf. Technol.*, vol. 29, pp. 16411–16439, 2024. doi: 10.1007/s10639-024-12499-0

[51] S. Criollo-C, J. E. Cerezo Uzcátegui, A. Guerrero-Arias, T. A. Yánez, A. D. Samala, S. Rawas, and S. Luján-Mora, "Use of virtual reality as an educational tool: A comparison between engineering students and teachers," *IEEE Access*, vol. 12, pp. 86662–86674, 2024. doi: 10.1109/ACCESS.2024.3416673

[52] J. Dewey, *Experience and Education*, New York, NY, USA: Touchstone, 1997.

[53] K. Lewin, *Field Theory in Social Science*, London, U.K.: Harper & Row, 1951.

[54] D. A. Kolb, *Experiential Learning: Experience as the Source of Learning and Development*, Englewood Cliffs, NJ, USA: Prentice Hall, 1984.

[55] T. H. Morris, "Experiential learning—A systematic review and revision of Kolb's model," *Interact. Learn. Environ.*, vol. 28, no. 8, pp. 1064–1077, 2020. doi: 10.1080/10494820.2019.1570279

[56] E. Krokos, C. Plaisant, and A. Varshney, "Virtual memory palaces: Immersion aids recall," *Virtual Reality*, vol. 23, no. 1, pp. 1–15, Mar. 2019. doi: 10.1007/s10055-018-0346-3

[57] B. Wu, X. Yu, and X. Gu, "Effectiveness of immersive virtual reality using head-mounted displays on learning performance: A meta-analysis," *Brit. J. Educ. Technol.*, vol. 51, no. 6, pp. 1992–2005, 2020. doi: 10.1111/bjet.13023

[58] Z. Yu, "A meta-analysis of the effect of virtual reality technology use in education," *Interact. Learn. Environ.*, vol. 31, no. 8, pp. 4956–4976, 2023. doi: 10.1080/10494820.2021.1989466

[59] R. Villena-Taranilla, S. Tirado-Olivares, R. Cózar-Gutiérrez, and J. A. González-Calero, "Effects of virtual reality on learning outcomes in K–6 education: A meta-analysis," *Educ. Res. Rev.*, vol. 35, 100434, 2022. doi: 10.1016/j.edurev.2022.100434

[60] M. Coban, Y. Islam, and B. I. Goksu, "The potential of immersive virtual reality to enhance learning: A meta-analysis," *Educ. Res. Rev.*, vol. 36, 100452, 2022. doi: 10.1016/j.edurev.2022.100452

[61] G. Makransky, T. S. Terkildsen, and R. E. Mayer, "Adding immersive virtual reality to a science lab simulation causes more presence but less learning," *Learn. Instr.*, vol. 60, pp. 225–236, Apr. 2019. doi: 10.1016/j.learninstruc.2017.12.007

[62] L. Cohen, L. Manion, and K. Morrison, *Research Methods in Education*, 8th ed., London, U.K.: Routledge, 2018.

[63] J. Brooke, "SUS: a retrospective," *J. Usability Studies,* vol. 8, no. 2, pp. 29–40, Feb. 2013. doi: 10.5555/2817912.2817913

[64] D. Kamińska, G. Zwoliński, and A. Laska-Leśniewicz, "Usability testing of virtual reality applications—The pilot study," *Sensors*, vol. 22, no. 4, p. 1342, 2022. doi: 10.3390/s22041342

[65] J. R. Landis and G. G. Koch, "The measurement of observer agreement for categorical data," *Biometrics*, vol. 33, no. 1, pp. 159–174, PMID: 843571, 1977.

[66] A. F. Hayes and J. J. Coutts, "Use omega rather than Cronbach's alpha for estimating reliability. But…," *Commun. Methods Meas.*, vol. 14, no. 1, pp. 1–24, 2020. doi: 10.1080/19312458.2020.1718629