VIRTUAL INDUSTRIAL ROBOT

CST6-24

Department of Industrial Engineering

Study Leader: Dr Clint Alex Steed

28-10-2024

Meyer van Dyk 22623531

University of Stellenbosch

Acknowledgements

I would like to express my sincere gratitude to all those who contributed to the development of this research project.

First and foremost, I would like to thank my supervisor, Dr Steed, for his continued guidance, support, and constructive feedback throughout the entire project.

I would also like to acknowledge the contributions of my fellow students and participants in the pilot study for their willingness to engage with the learning app and provide insightful feedback.

Special thanks to Stellenbosch University’s Industrial Engineering department for providing the necessary resources and facilities to conduct this research.

Lastly, I would like to express my heartfelt thanks to my family and friends for their unwavering support and encouragement throughout this journey.

Plagiarism Declaration

UNIVERSITEIT·STELLENBOSCH·UNIVERSITY
jou kennisvennoot • your knowledge partner

1. Plagiarism is the use of ideas, material and other intellectual property of another's work and to present is as my own.
2. I agree that plagiarism is a punishable offence because it constitutes theft.
3. I also understand that direct translations are plagiarism.
4. Accordingly, all quotations and contributions from any source whatsoever (including the internet) have been cited fully. I understand that the reproduction of text without quotation marks (even when the source is cited) is plagiarism.
5. I declare that the work contained in this assignment, except where otherwise stated, is my original work and that I have not previously (in its entirety or in part) submitted it for grading in this module/assignment or another module/assignment.

Student Number	**Signature**
22623531	
Initials and Surname	**Date**
M van Dyk	28-10-2024

Graduate Attributes

Graduate Attribute	***Section***	***Description***
1->Problem Solving	Entire document	Problem was defined, analysed and solution objectives were established. Literature review provided insight and tools to develop solution. Solution was evaluated in a pilot study and results were reviewed and conclusions were drawn.
5->Engineering Methods, Skills and Tools	3. Methodology 4. Results	Project planning to create milestone timelines. Google sheets to store result data. Git-hub to create version control alongside an online backup. Various models and frameworks to ensure structured and planned development of solution. Results were evaluated and discussed using statistical analysis.
6->Professional and Technical Communication	Entire document Oral Examination	Appropriate structure, style and language for target audience supported by graphical support used throughout report. Oral Examination to assess effective oral communication.
9->Independent Learning Ability	2. Literature Review 3. Methodology 5. Discussion and Recommended Improvements 6. Conclusion	Conclusion features the reflection of own learning requirements and discusses implications of solution. Information was sourced and evaluated to form solution and tools used to develop and evaluate solution. UNITY3D application alongside native coding skills were developed to execute solution alongside introductory 3D modelling skills and introductory network communication skills.
10->Engineering Professionalism	3. Methodology 5. Discussion and Recommended Improvements 8.6 Appendix E: Project Plan	Decision making judgement demonstrated in methodology using insights gained from literature review and objectives of solution. Boundaries of competence and practical aspects related to time frame discussed alongside awareness of requirements to stay up-to date.

Abstract

Industrial robots have been crucial in the manufacturing industry although adoption rates are slow, especially in South Africa due to financial and social factors. One of the factors include the lack of skilled workers available to operate and utilize Industrial Companion Robots (ICR). The Industrial Engineering faculty of Stellenbosch University has recognized the need for ICR programming to be introduced into the curriculum. Although with the high cost of ICRs, it is infeasible to buy additional robots, causing a practical and logistical problem. Currently learning resources are not suited for in-depth classroom use, while high-fidelity simulation environments though powerful, are overly complex for practical classroom application and the required learning objectives.

This study proposed a solution to the problem by developing a simulation-based virtual tool to solve this problem. The proposed simulation tool was developed with considerations from educational frameworks to ensure it meets pedagogical standards and effectively supports learning. The tool utilizes the industry grade robot controller URSim, developed by a leading companion robot manufacturer Universal Robots. A Unity3D application was created that receives input from the robot controller to control a virtual robot model to complete a series of tasks within the virtual environment. The controller is run on a windows laptop utilizing a virtual Linux machine, communicating with the educational tool run on an Android tablet. A pilot study was conducted to evaluate the educational tool's effectiveness and gather feedback from participants. The results of the pilot study indicate that this proof-of-concept application effectively satisfies objectives to offer a scalable and cost-effective alternative to education requiring physical models without sacrificing practical components of the learning material.

Table of Contents

Table of Figures

Table of Figures

Glossary

- Bug(s): In computer technology, a bug is a coding error in a computer program.
- Cyber-Physical Systems: Systems that integrate computational algorithms with physical processes, enabling enhanced monitoring and control.
- Industrial Companion Robots (ICR) and cobots: Terms used interchangeably to describe robots designed to work alongside humans in a shared workspace.
- Internet of Things (IoT): A network of interconnected devices that collect, exchange, and act upon data.
- Pedagogically: In a manner that relates to teaching or education, especially in terms of methods and principles.
- Robotics: The technology and design of robots for performing tasks traditionally done by humans.
- Smart Factories: Advanced manufacturing facilities that use digital technology to improve production efficiency and flexibility.
- TCP (Tool Centre Point): The specific point on the end effector or tool of a robot, such as a gripper or sensor, used for positioning and orientation control. It represents the exact location in space where the robot interacts with its environment, and all movements are calculated relative to this point.

- VMs (Virtual Machines): Isolated virtual environments created by virtualization software that allow for the running of multiple operating systems and applications on a single physical computer.
- Virtual Environments (VE): Simulated digital spaces where users can interact with a computer-generated environment, often used for training, education, or entertainment purposes.

Chapter 1 - Introduction

1.1. Background

The Industrial Engineering department owns a single Industrial Companion Robot (ICR), commonly used in modern engineering companies and on production lines and aims to provide undergraduate students with practical exposure to the robot and its programming. However, logistical and practical challenges arise due to the limited availability of the ICR—there is only one robot for a class of approximately 120 students. This imbalance makes it impractical to allow each student sufficient hands-on experience with the robot and its accompanying software. Furthermore, the high costs associated with ICRs, which typically range from R500,000 and upwards (Universal Robot Cobot Cost Analysis and Return on Investment: Electromate Inc), make purchasing additional units unfeasible. These costs do not include installation, maintenance, and essential accessories such as grippers.

While some ICR manufacturers provide e-learning resources, such as the Universal Robots Academy, these courses are often lengthy, overly basic, and not well-suited for in-class use. As a result, they fail to adequately meet the educational needs of undergraduate engineering students, particularly those who require more in-depth, practical learning experiences.

An alternative to direct interaction with the ICR is the use of simulations. High-fidelity simulation environments, such as NVIDIA Isaac Sim and CoppeliaSIM, are widely used in ICR programming and industrial applications. However, these software packages come with their own challenges. They typically require extensive training and several hours of use before students can produce meaningful results (CoppeliaSIM User Manual). Additionally, these simulation environments are designed to simulate entire processes rather than focusing on the detailed control and operation of individual ICRs, making them overly complex for classroom use and out of scope for the desired learning material. Given these limitations, there is a clear need for a more accessible educational solution that enables students to gain practical experience with ICRs in a classroom setting. This project proposes the development of a simulation tool to bridge this gap, offering students exposure to ICR programming in a controlled virtual environment.

1.2. Problem Statement

With the increasing integration of industrial companion robots in modern manufacturing environments, particularly in the context of Industry 4.0, there is a growing demand for engineers who are proficient in both the theoretical understanding and practical programming of these robots. However, the current engineering curriculum lacks sufficient hands-on training with advanced robotics systems, leaving students ill-prepared to meet the needs of the evolving workforce. The limited availability of industrial robots for training purposes and cost concerns, further restricts opportunities to practical learning opportunities.

This paper aims to address this educational gap by developing a proof-of-concept educational tool that allows undergraduate engineers to learn how to operate and program industrial companion robots in a scalable, cost-effective, and engaging environment. The tool, designed as a serious game-like application, provides an interactive learning experience by emulating real-world robotics tasks. While not intended as a final product for educational use, this project serves as a conceptual model to demonstrate the potential of virtual tools in bridging the gap between theoretical robotics education and the practical skills required for the next and current generation of engineers.

1.3. Objectives

- Develop a Virtual Companion App: Create a simulation-based application using a game engine that replicates the operation and programming of an Industrial Companion Robot (ICR).
- Facilitate Remote Access and Control: Enable students to connect from virtual machines to control the simulated ICR, providing hands-on experience in a virtual environment.
- Design Educational Challenges: Create a series of interactive challenges or problems that progressively develop students' skills in programming and operating ICRs.
- Bridge the Learning Gap: Address the lack of classroom-optimized tools by offering a user-friendly, accessible alternative to high-fidelity industrial simulations, which are not tailored for classroom use.
- Evaluate Learning Outcomes: Assess the effectiveness of the virtual companion app in improving students' understanding of ICR programming and operation through feedback and performance metrics.
- Proof of Concept: Demonstrate the feasibility of using a simulation as a supplemental educational tool, rather than a replacement for real ICR experience.

1.4. Scope

- Develop a Virtual ICR Companion App: Design and create a virtual Industrial Companion Robot (ICR) app using a game engine, to simulate ICR operation and programming.
- Test and Validate the Virtual ICR: Ensure the functionality of the virtual ICR and companion app, assessing its accuracy, responsiveness, and relevance to real-world ICR operation.
- Create User-Friendly Instructional Guides: Develop clear and structured instructions for users/students to learn how to interact with the virtual ICR and companion app, ensuring ease of use.
- Design Interactive Challenges: Implement a series of progressively complex challenges or tasks aimed at testing students' comprehension of ICR programming and operation.
- Gather Feedback and Assess Learning Outcomes: Use surveys and assessments to evaluate the students' experience, gather feedback on usability and learning outcomes, and analyse the effectiveness of the companion app.

1.5. Robotics versus Industrial Robotics

The theory and design principles of robotics consists of topics such as forward and inverse kinematics, manipulator dynamics, and singularities (Craig, 2005). These principles are foundational for robotic design and operation. However, this project focuses specifically on addressing the educational gap in programming Industrial Companion Robots, simplified robotic solutions for industrial tasks. Unlike roboticists, users of ICRs do not require in-depth knowledge of complex robotic theory to operate or program these tools. Manufacturers such as Universal Robots have developed user-friendly interfaces, such as Polyscope, to make ICR programming accessible.

For this reason, a more appropriate framework is presented by Groover. Groover discusses industrial robotics in terms of anatomy, control systems, and programming within industrial environments (Groover, 2007). In practice, engineers and technicians often use manual leadthrough programming, where the robot is physically guided through its intended path. However, this method is unsuitable for a virtual learning environment. To address this, the educational tool replicates offline motion programming via Polyscope's graphical interface. This allows users to test and develop robot programs virtually without disrupting real-world operations, making it an effective tool for bridging the gap between theoretical concepts and practical skills in industrial robotics.

Chapter 2 - Literature Review

The rapid advancement of industrial automation, particularly through the implementation of Industry 4.0, has significantly reshaped manufacturing and engineering practices. Industrial robots, including Industrial Companion Robots, are central to this transformation. This literature review explores the evolution of industrial robots within the context of Industry 4.0, their benefits, and their adoption in South Africa. Additionally, various educational methodologies, such as the ADDIE model and NASA TLX, are analysed to provide insight into how simulation-based learning can be employed to develop effective educational tasks. These methodologies focus on designing activities that progressively build students' knowledge and skills in ICR programming and operation, fostering deeper engagement and practical application.

2.1. Industrial Robots

Industrial robots have become a cornerstone of modern manufacturing, driving efficiency, precision, and flexibility in production processes. Their integration is a key component of Industry 4.0, the latest industrial revolution that combines advanced automation, data exchange, and smart technologies. This section explores the role of industrial robots in Industry 4.0, with a particular focus on the emergence of Industrial Companion Robots. By examining the adoption of these technologies in South Africa, this section provides insight into their significance and potential impact on the future of industrial automation.

2.1.1. Industry 4.0

Industry refers to the sector of an economy focused on producing goods through mechanized and automated processes (Lasi *et al.*, 2014). Technology plays an integral part in driving paradigm shifts, as innovation and advancements have continually redefined industrial practices since the onset of industrialization. The first industrial revolution or ‘Industry 1’ was driven by mechanization, then the industrial use of electrical energy powered the second revolution. The creation and adoption of the internet and computers led to the digitalization of almost every aspect of human life including industry in its so-called third revolution.

Following the advancements of previous industrial revolutions, The German government publicly introduced the concept of the fourth industrial revolution (a.k.a. Industry 4.0) in 2011(Xu *et al.*, 2021). Industry 4.0 integrates cyber-physical systems, the Internet of Things

(IoT), robotics, and smart factories to enable real-time communication and automation in manufacturing processes and supply chains (Xu *et al.*, 2021).

This paradigm shift allows for increased efficiency, flexibility, and data-driven decision-making in industrial settings (Lasi *et al.*, 2014). Companies are required to adapt their business models and operational structures.

2.1.2. Industrial Robots in Industry 4.0

Robotics is a crucial component within Industry 4.0, which encompasses a variety of technologies that drive advancements in manufacturing. The integration of robotics into Industry 4.0 facilitates enhanced automation systems, providing significant benefits in the manufacturing sector (Goel and Gupta, 2019).

One of the primary advantages of robots is their ability to perform repetitive tasks with greater precision and at a lower cost compared to human workers. By taking over these routine tasks, robots allow human employees to engage in more complex and rewarding activities. This shift not only improves job satisfaction but also increases the variety and scope of their work, contrasting sharply with the monotony of previous roles.

Robots also offer the advantage of continuous operation, working tirelessly without breaks. This leads to increased production rates and reduced downtime compared to human workers (Goel and Gupta, 2019). Additionally, robots provide a level of consistency and precision that surpasses even the most experienced human workers (Javaid *et al.*, 2021), minimizing variations in product quality and reducing waste throughout the manufacturing process.

The implementation of Industry 4.0 relies heavily on real-time data collection and analysis. Robots, equipped with advanced sensors and data collection capabilities, enable continuous monitoring and optimization of manufacturing processes (Javaid *et al.*, 2021). This real-time analysis facilitates better decision-making, shortens production cycles, and improves lead times, ultimately enhancing production capabilities.

While the initial investment in robotic technology can be significant, the long-term benefits—such as increased efficiency, reduced labour costs, and lower waste—often justify the expenditure. The overall improvements in manufacturing processes lead to substantial cost savings and enhanced production capabilities (Goel and Gupta, 2019).

2.1.3. Industrial Companion Robots

Traditional robots in manufacturing operate in isolated sections and are usually separated from human workers for safety reasons. A typical application of these robots are assembly lines performing repetitive tasks and processes i.e. a bottling plant robot placing caps on bottles. Collaborative robots are transforming the industrial automation workplace. ICRs are designed to operate safely and in direct interaction with human workers, unlike traditional industrial robots and machines. This integration between man and machine offers several advantages.

The ICRs in a BMW factory alongside human workers significantly improved productivity when compared to teams of only robots or only human teams. MIT researchers found that ICRs are perfect at repetitive tasks which allows human workers to focus on more complex activities. The ICRs reduced human idle time by 80% (Unhelkar et al., 2018).

ICRs are typically smaller and more lightweight than traditional robots which allows them to operate close to humans without a risk of injury (Colgate et al., 1996). Many ICR manufacturers integrate sensors to detect a possible collision with accompanying code to stop an ICR if the need arises. Another safety feature is force-limiting technology to decrease injury if sensor-based collisions fail. These features allow ICRs to reduce workplace accidents and be used in tandem with humans safely (Shin et al., 2019).

ICRs are generally less expensive (*Universal Robot Cobot Cost Analysis and return on Investment: Electromate Inc*) than traditional robots due to the small design and setup requirements. Traditional robots are usually designed for one specific task and are limited by trained operators and maintenance services/employees (Faccio, Bottin and Rosati, 2019). ICRs can perform a wide range of tasks with a wide range of accessories (such as grippers, drills etc.) which makes them a powerful and adaptable tool in the workplace. ICR programming has a steep learning curve although a trained operator can set up different processes and protocols with great ease. This adaptability and relative ease of use make ICRs cost-effective when compared to traditional industrial robots.

An often overlooked but important aspect is the effect on employee morale. ICRs often take over repetitive and potentially hazardous tasks which free up human workers for more stimulating and meaningful activities. This often increases employee morale, and job satisfaction and increases workers' skills.

There is a growing desire and interest in the implementation of cobots in various industries although, there is a lack of knowledge as to how to optimally utilize and implement these robots in manufacturing processes. It has been found that cobots are better suited in low-volume and high-variability tasks although, they lack in reliability, precision and productivity when compared to traditional robots in repetitive tasks (Ibn Tofail University, National School of Applied Sciences, Engineering Sciences Laboratory, Kenitra, Morocco *et al.*, 2024).

2.1.4. Overlook of Industrial Companion Robots in South Africa

The adoption of industrial collaborative robots in South Africa, especially the Western Cape, is rising although the environment is still in an early stage of development. There is however a promising potential to utilize ICRs in industry.

The South African government recognized the advantages and role of robotics in industrial growth. The National Development Plan 2030 states the importance of automation and skills development in the future workforce (*National development plan 2030*). ICR as stated above allows workers to focus on more complex tasks and less repetitive tasks, upgrading the workers to a higher skill set. South Africa suffers from skilled labour shortages (Maisiri and van Dyk, 2019) thus ICRs can free up skilled workers to allocate more time to higher-value activities. Productivity increases will allow for business success in South Africa due to the previously mentioned advantages of employee utilization. Increased productivity will subsequently lead to better profitability which is crucial in a developing country.

Safety is an important aspect of any industrial process, and South African manufacturers are no exception. Providing the safety features of ICRs, the workplace injury rate when compared to an exclusively human workforce or a mix of traditional robots is also an important aspect. This could save enterprises money as there is less production lost to injuries, costs due to injury and reduce employee turnover as they perform safer and more satisfying work.

There are many advantages and potential for the Western Cape and South Africa to adopt ICRs in manufacturing, although some challenges and considerations are present. ICRs are cost-effective in comparison to traditional robots although require substantial capital and investment, especially in the context of small and medium enterprises (SMEs) in South Africa (Mpungose, S.C). In addition to the capital investment, the workforce would need upskilling in the areas of programming, maintenance, and robot collaboration. A crucial requirement for successful integration of human-robot workspaces.

A robust ICR ecosystem in South Africa requires investment in infrastructure for training, maintenance and support services (Chigbu and Nekhwevha, 2022), an ecosystem that is still developing as there is only one official distributor (*Find your universal robots distributor, universalrobots.com*) according to Universal Robots, a leading manufacturer of ICRs.

A study done in the automobile industry in South Africa (Chigbu and Nekhwevha, 2022), found that robots and human employees work efficiently together and produced higher product quality and throughput. It was also found that collaborative work between human employees and robots resulted in negative impacts on job satisfaction and confidence of autoworkers. A large pushback from the African workforce is prevalent due to the African culture and job-scarcity in South Africa (*The future African workplace: the use of collaborative robots in manufacturing*).

SMEs in South Africa are generally aware of global trends in Industry 4.0 and elements such as cobots. Several factors provide extreme resistance to the adoption of cobots in SMEs. The factors include financing, skill acquisition (in operation and implementation), information scarcity on robotics, employment regulations and pushback from the African workforce.

2.1.5. Conclusion of Industrial Robots

The exploration of industrial robots within the context of Industry 4.0 highlights their potential in transforming manufacturing processes. Industrial robots have been instrumental in enhancing efficiency, precision, and productivity, offering significant advantages over human labour in repetitive and hazardous tasks. The integration of robots into Industry 4.0 has further amplified these benefits, driving innovation through real-time data collection, process optimization, and continuous operation.

ICRs, with their ability to work alongside human operators, represent a notable advancement in robotic technology. Unlike traditional robots that operate in isolation, ICRs enhance human-robot collaboration by combining the strengths of both human intuition and robotic precision. This integration has shown to improve productivity, safety, and employee satisfaction. Despite the challenges associated with the initial investment and the need for skilled personnel, the advantages of ICRs, including increased safety and adaptability, present a compelling case for their adoption.

In South Africa, while the potential benefits of ICRs are recognized, including increased productivity and improved safety, the challenges such as capital investment, skill gaps, and resistance from the workforce remain. Addressing these challenges is crucial for harnessing the full potential of ICRs and achieving sustainable growth in the South African manufacturing sector.

Overall, the progression from traditional industrial robots to advanced ICRs under Industry 4.0 marks a significant shift in manufacturing practices, underscoring the need for skilled workers and adaptation to fully leverage the capabilities of modern robotics.

2.2. Lens of Learning

The effective design of the Virtual Environment (VE), where students learn to operate an Industrial Companion Robot, is critical to the project's success. This project seeks to address the shortcomings of existing classroom-based educational and training resources for ICR programming and operation. To achieve this, the design of educational tasks within the VE must be grounded in well-established learning methodologies. These methodologies guide the creation of instructional materials and assessments, ensuring that students gain a meaningful understanding of ICR programming and control. The following section explores relevant educational frameworks that inform the design of this learning experience.

2.2.1. Tait’s Model

Alan Tait proposes a cognitive, affective, and systematic model to design effective learning experiences (Tait, 2000). A learner’s cognitive processes include comprehension, problem-solving, and critical thinking (Tait, 2000). To fulfil a learner’s cognitive needs effectively, mechanisms that support this process should be clear to avoid potential confusion. A consistent structure in instructional materials is essential to avoid distracting the learner from the educational material. Active engagement opportunities with the educational content are also essential to form and optimise cognitive processes when a learner interacts with new content. Scaffolding is a method where learners are faced with progressively complex tasks to complete, providing feedback at timely moments to help reflect and solidify new learning material.

New educational content should be balanced in difficulty to provide a positive learning experience as a learner's emotional and motivational need impact their experience and retention (Tait, 2000). The importance of a reasonable level of difficulty is to ensure cognitive processes are exercised during the session although if it is perceived as too difficult, feelings of anxiety and self-doubt can interfere with the content and cause a negative feeling towards the new material. Affective support mechanisms are essential to this balance to motivate and assist the learner by encouraging social interaction through peers and empathetic instructors. Successful balance of affective support and cognitive stimulus positively impacts learner engagement and retention.

Clear communication channels should be present in the activity to provide systematic support for the learner (Tait, 2000). Consistency, reliability, and accessibility of support services is essential to provide confidence to a learner to seek assistance and effectively complete the learning process. The three principles of cognitive processes, affective support and systematic support are not independent and should be aligned to meet learner's needs holistically to provide an effective learning environment. Although the design of such learning environments is continuous and adaptive. Effective learning environments requires an iterative process which reflects on feedback from students and their assessment results.

2.2.2. ADDIE Instructional Model

The ADDIE instructional model is a tool used to develop educational and training programs (Ganesan, 2015). The steps of the model are as follows:

- <u>Analysis</u> of the requirements, task, and audience.
- <u>Design</u> of learning objectives, outline, and delivery media.
- <u>Development</u> of graphics, storyboards, and construction of learning materials.
- <u>Implementation and Evaluation</u> through quality tests and result evaluation. (Ganesan, 2015)

A common approach to apply the model is the use of a table populated with each phase and their respective inputs, tasks, and outputs. The ADDIE model can be used to develop the educational tool to ensure the design and effective completion of learning objectives. This contributes to the crucial objective of designing and developing not only the robot learning app but also its educational challenges.

Virtual educational tools have been developed using the ADDIE model and has produced successful educational tools. An example is the Virtual Laboratory (VLab) developed by the school of Engineering in Aberdeen, UK. The purpose of the VLab was to create a safe environment to facilitate virtual experiments in Science, Technology, Engineering, and Mathematics (STEM). VLab was developed with the ADDIE model and produced results that indicated an effective, collaborative and interactive learning environment (Amish and Jihan, 2023).

2.2.3. NASA Task Load Index

A popular tool for measuring subjective mental workload is the NASA Task Load Index (TLX). It relies on a multifaceted approach to determine the weighted average of subdimensions of mental demand, physical demand, temporal demand, performance, effort, and frustration level. These subdimensions are rated from “Low” to “High” by subjects after completing a task. Each subdimension are assigned weights from 0 to 6 to indicate the least and most relevant subdimension in the evaluation. The weighted average for the overall workload score is calculated from each subject’s ratings. (Cao *et al.*, 2009).

With supplementary questions, TLX can reveal the sources of perceived difficulties during a task. Researchers at Colby College, Waterville, USA, identified methods for educators to identify at-risk students and increase accessibility of difficult projects while retaining quality (Al Madi, Peng and Rogers, 2022) in introductory computer science modules. TLX provides educators with a powerful tool to not only receive feedback on tasks and projects, but also the ability to discover the exact elements of the task that cause negative experiences during teaching activities. The ease of use of the NASA TLX method and detailed analysis possible due to its multidimensionality has made it a popular choice in workload measurement (Cao *et al.*, 2009). The NASA TLX can be used to capture the perceived workload and demand of the educational challenges in the learning environment to ensure that challenges are well balanced in terms of difficulty, workload and demand of users.

2.2.4. Revised Bloom’s Taxonomy

Bloom’s Taxonomy is a tiered model of six cognitive levels that classify a subjects’ mastery of learning material (Forehand, M.,2010). The cognitive level’s build on one another, therefore if a student has achieved the 3rd level, they have already mastered the first two levels. Lorin Anderson, a former student of Bloom, revised the model to add relevance for 21st century students and educators (Forehand, M.,2010). Retaining 6 levels, the new version consists of verbs rather than nouns when describing the level of mastery a student has achieved.

The six major categories of the Revised Bloom's Taxonomy (RBT) (Forehand, M.,2010), in increasing order of mastery:

- Remembering – recalling relevant knowledge from long term memory.
- Understanding – constructing meaning through interpretating and explaining.
- Applying - using a procedure through executing or implementing.
- Analysing – dividing material into parts and the connection of parts and to an overall system.
- Evaluating – formulating judgements using criteria to critique.
- Creating – adding elements together to form a functional whole through generating, planning, or producing.

The accurate measurement of a student's ability and mastery of a subject is an important aspect of education. Using the RBT tool allows educators to achieve this by designing educational content with each level in mind and the objectives of learning.

2.2.5. Positive and Negative Affect Schedule

Providing students with an adequate feedback system is essential to record accurate results of the learning activity to measure success and determine if and which adjustments should be made to improve the overall effectiveness of the activity. The Positive and Negative Affect Schedule (PANAS) survey is a commonly used tool to assess the psychological effects of an activity on an individual.

The positive affect scale consists of items that assess the positive emotional effects an activity has on an individual (Watson et al., 1988). These effects range from contentment to joy and enthusiasm. The individual then rates their experienced emotions during the activity on a defined scale that usually ranges from "not at all" to "extremely". On the same rating scale, the individuals give feedback on the negative affect scale that range from anxiety, anger, and frustration.

The PANAS survey aims to provide feedback on the breadth and intensity the pair of affect scales the individual experienced during the activity. A slightly modified PANAS survey is an ideal tool to measure the affective and systematic support systems in place during the learning activity and gauge the emotional states experienced by learners.

2.2.6. Conclusion of Lens of Learning

This section outlines key educational frameworks and tools essential for designing an effective learning environment for operating ICRs.

Tait's Model emphasizes the need for a balanced approach that integrates cognitive, affective, and systematic support to create a comprehensive learning experience. This approach ensures that students are not only intellectually engaged but also emotionally supported, which is crucial for effective learning.

The ADDIE Instructional Model provides a structured methodology for developing educational programs. Its iterative phases—Analysis, Design, Development, Implementation, and Evaluation—ensure that instructional materials are well-crafted and responsive to student needs. The successful application of this model in developing virtual educational tools, demonstrates its efficacy in creating interactive and collaborative learning environments.

The NASA Task Load Index (TLX) offers a valuable mechanism for assessing the subjective mental workload of students. By evaluating different dimensions of workload, TLX helps educators identify potential challenges and areas for improvement, thereby enhancing the learning experience and ensuring that educational tasks are appropriately challenging.

Revised Bloom's Taxonomy provides a hierarchical framework for categorizing learning objectives and assessing student mastery of material. By aligning instructional content with this taxonomy, educators can effectively measure and support student progress across various levels of cognitive engagement.

Finally, the Positive and Negative Affect Schedule (PANAS) is instrumental in capturing the emotional impact of the learning experience. By assessing both positive and negative affective responses, PANAS helps in evaluating the overall effectiveness of the educational tasks and ensuring that they foster a supportive and engaging learning environment.

Overall, integrating these educational methodologies and tools ensures the development of a robust and effective Virtual Environment (VE) for teaching ICR programming. This holistic approach addresses the current limitations of classroom-based resources and provide a comprehensive learning experience that is both informative and supportive.

2.3. Overall Conclusion of Literature Review

The literature review highlights the transformative impact of robotics within the context of industrial evolution, particularly under Industry 4.0. The progression from traditional industrial robots to advanced ICRs illustrates a significant shift in manufacturing practices, driven by enhanced automation, real-time data analysis, and improved human-robot collaboration. ICRs, which operate alongside human workers, not only boost productivity and safety but also address the limitations of traditional robots by offering greater flexibility and adaptability in various manufacturing tasks.

In South Africa, the adoption of ICRs aligns with the National Development Plan 2030 and presents a promising opportunity to enhance productivity and safety in the manufacturing sector. However, challenges such as high initial investment, skill gaps, and workforce resistance must be addressed to realize these benefits fully. A robust ecosystem for ICR integration requires substantial investment in training, maintenance, and support infrastructure.

Given these considerations, it is crucial to develop educational tools and methodologies that effectively prepare students in South Africa for the evolving demands of the robotics industry. By leveraging well-established learning frameworks, such as Tait's Model, the ADDIE Instructional Model, and Revised Bloom's Taxonomy, alongside tools like the NASA Task Load Index and the Positive and Negative Affect Schedule, the design of educational tasks and activities can be optimized to bridge the gap between classroom learning and real-world applications.

The effective design of a Virtual Environment for learning ICR programming and operation is central to this educational endeavour. The project aims to address existing gaps in classroom-based resources by providing an interactive, hands-on learning experience that reflects the complexities of modern industrial settings. This approach not only enhances students' understanding and skills in robotics but also contribute to addressing the skill shortages and fostering a more skilled workforce in South Africa's manufacturing sector.

In summary, the integration of ICRs into manufacturing practices, supported by targeted educational initiatives, represents a significant step forward in advancing industrial capabilities and addressing skill gaps in South Africa. The project’s focus on developing a Virtual Environment for ICR education is a strategic response to these needs, aiming to equip students with the knowledge and skills necessary to thrive in an increasingly automated and data-driven industrial landscape.

Chapter 3 - Methodology

This section outlines the approach and methods used to develop and assess the companion app as an educational tool for undergraduate engineering students. The companion app aims to teach students about cobots, the methodology is structured to ensure the app is effective, engaging, and capable of meeting the project's objectives.

The process began with the design and development of virtual environments in Unity. These environments are crafted to simulate real-world scenarios where students can interact with cobots. The design incorporates principles from established educational frameworks to ensure that the virtual tasks are both instructional and challenging.

Following development, the functionality of the virtual environments was tested using the associated educational tools. This involved conducting a study with students to gather feedback on their experience and identify areas for improvement. The evaluation phase also included assessing how well the app facilitates learning and skill development while determining the perceived task demand and challenges. The overall methodology is shown in Figure 1: Methodology Flow Diagram

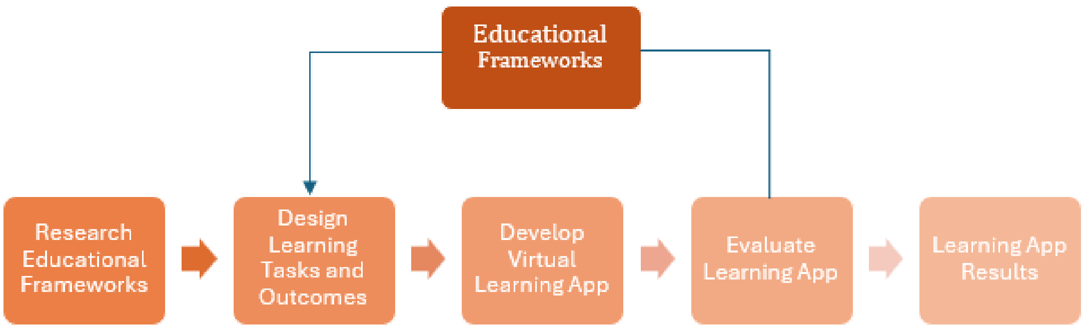

Figure 1: Methodology Flow Diagram

3.1. Foundation and Contributions of the Project

The foundation of this project is built upon the Unity3D model developed by Roman Parak, a researcher based in Brno, Czech Republic. Mr. Parak's contributions include the creation of a detailed Unity3D model of the UR robot and the establishment of the connection between this model and URSim. This work serves as a critical starting point for the project. Leveraging Mr. Parak's model, this project aims to extend its application by developing educational tools and virtual environments specifically designed to test and instruct students. This study focuses on adapting and expanding the virtual environment of the model to create a robust educational framework that addresses current gaps in ICR training and enhances the learning experience for students. The original project and model by Mr Parak can be found here: https://github.com/rparak/Unity3D_Robotics_UR

3.2. Technical System Overview

The first step is to create a virtual ICR and its environment using the Unity game engine to control the ICR and facilitate the scaffolding challenges a learner must complete. The app must then be connected to a virtual machine software VMware workstation player. The virtual machine is used to run URSim, software that simulates an ICR. The virtual environment provides the user with real-time visual feedback while the simulation software simulates the movement of a real ICR with communication between the programs on the position and movements of the ICR. A visual overview of the system is presented in Figure 2: Overview of System.

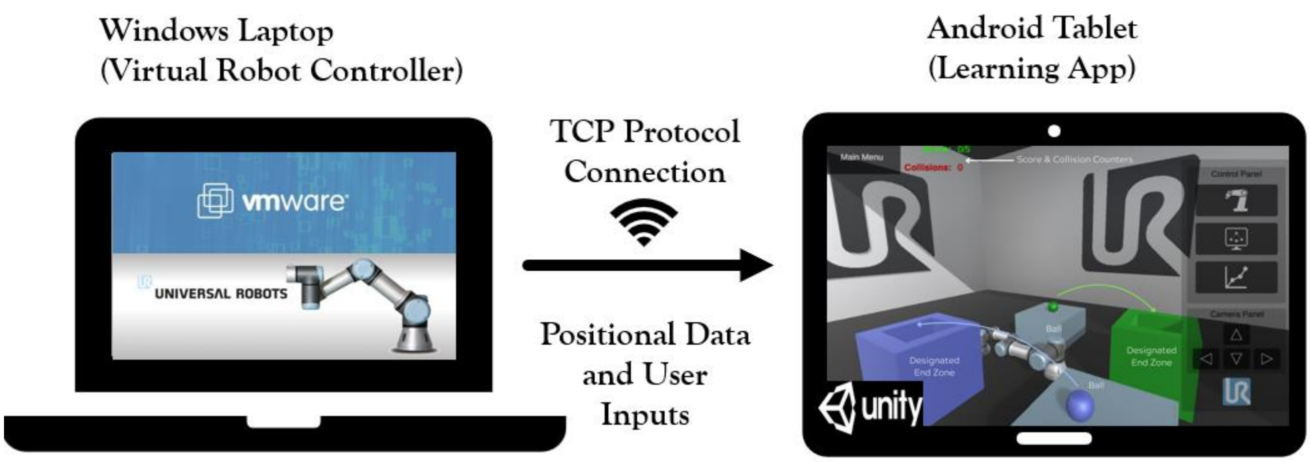

Figure 2: Overview of System

There are three applications that interact to create a virtual cobot and the virtual educational environment:

- Unity game engine
- URSim robot controller simulation software
- VMWARE virtual machine software

3.2.1. Unity Game Engine

Unity is a versatile and widely used game engine that provides a comprehensive development environment for creating both 2D and 3D applications using the C# coding language. It allows developers to design and implement interactive experiences through a robust set of tools and features. In this project, Unity serves as the foundational platform for developing virtual environments to facilitate the education and training of students in operating cobots. By leveraging Unity's capabilities, we can create interactive tests, enabling students to practice and refine their skills in a controlled and engaging setting. The virtual environments developed in Unity are designed to mimic common pick-and-place tasks performed by cobots, offering students hands-on experience with cobots without the constraints and risks of a physical environment and necessity of a physical robot. This approach not only enhances learning outcomes but also provides a scalable and adaptable solution for training across various educational contexts. It is recommended that an Android tablet is used to run the application created in unity. The use of a tablet enables students to use their laptops for programming and control of the robot while receiving visual feedback from the tablet although this is optional as the application can also be run on a windows laptop (either the same laptop running the simulation software or a different laptop).

3.2.2. URSim Robot Simulation Software

URSim is a simulation software developed by Universal Robots that provides a virtual environment for programming and testing Universal Robots' cobots. It mimics the behaviour and functionality of the physical robots, allowing users to develop and debug robot programs without needing direct access to the hardware. In this project, URSim is utilized to complement Unity's virtual environments by providing a realistic and interactive simulation of cobot operations. By integrating URSim, students can gain practical experience with robot programming and control within a virtual space that accurately reflects the capabilities of the actual robots. This combination of Unity and URSim enables a comprehensive training experience, where students can experiment with and refine their skills in a simulated environment that mirrors real-world industrial applications. URSim runs on a Linux operating system, as most student use Windows operating systems, the use of a Virtual Machine is required.

Universal Robots is a leading manufacturer of collaborative robots and the developers of URSim. The use of URSim ensures that the educational tool developed is the same as used in industry, enhancing the effectiveness of the training and improving students' preparedness for real-world scenarios. Student will have the choice between manually controlling the robot with controls that move or rotate various joints of the simulate robot, although this is not often used in practice, or writing code to automate the robot's movements to perform repetitive tasks. The VE tasks are designed to force students to write code for the robot instead of manual operation.

3.2.3. VMware Virtual Machine Software

VMware is a virtualization platform that allows for the creation and management of virtual machines (VMs) on a single physical computer. It enables users to run multiple operating systems simultaneously on one machine, each within its own isolated environment. In this project, VMware is used to create and manage virtual machines that allow students to run URSim with their Windows laptops. This setup allows for a streamlined and controlled environment where all components of the project can operate independently yet cohesively. By using VMware, students can access and interact with the virtual environments and simulation tools without needing specialized hardware for each software component. This approach not only simplifies the deployment and management of the educational tools but also ensures that students can engage with the project's resources in a consistent and reliable manner, enhancing the overall learning experience.

3.3. Design of Virtual Environment

The virtual environment offers students an interactive and practical experience to develop their skills in programming and operating collaborative robots. Firstly, an overview of the virtual environment is presented, secondly the various tasks to be completed by the user is outlined which is then followed by the logic used in the creation of the objects and interactions that take place in the virtual environment.

3.3.1. Overview of Virtual Environment

The environment is structured into different stages, each presenting progressively complex tasks, allowing students to gradually build their understanding and proficiency. From the main menu, users can access various stages, including Freeplay, which provides a risk-free space for students to explore and become familiar with the robot's controls and features.

As this project serves as a proof of concept, no physical collision between the robot and the environment is implemented. However, to reinforce real-world safety considerations—where collisions could cause damage, injury, or even death—a collision penalty system is in place. This system detects when the robot collides with objects and tracks penalties to encourage students to avoid collisions, emphasizing the importance of safety in real-world applications.

The most common use of cobots in industry is pick-and-place tasks. These tasks entail the robot to pick up a specific object and place the object in a designated zone. This was the rationale for selecting this process as the main objective in each stage. The Tool Centre Point (TCP) of the robot refers to the specific point on the end of the tool (i.e. a gripper) that is used for positioning and orientation control of the robot. For simplicity and due to the scope of this project, the student must simply touch an object with the TCP to pick it up. If the student moves the object into the designated zone, the object will detach from the TCP and fall into a basket, indicating a successful pick-and-place process which will increase the score. When the required score for that stage is reached, the stage is complete, and the student can move on to the next stage.

3.3.2. Scaffolding Stages

To enforce the use of coding, each stage has a score system to force repetition of the task. The repetition will eliminate manual control as an option due to the tediousness and time required to manually control the robot. In the application, stages are referred to as "Levels". Images of each level are attached to appendix A for visualization purposes.

Level 1 requires the student to pick up a ball and place it in the designated zone. This will require the student to do simple linear pathing and teaches students the basics of robot movement. Level 2 is identical although the required score is increased to encourage coding.

Level 3 is the addition of multiple objects and designated zones. The student must insert each object into the corresponding zone. The objects and zones are colour co-ordinated to avoid confusion. This level reinforces the lessons learnt in the first two levels with added complexity. Level 4 is again identical to level 3 although requires repetition of the task. The score will only increase if both objects are in the correct corresponding zone. If a student places an object in the wrong zone, the object will return to its starting position. Level 4 is shown in Figure 3: Level 4.

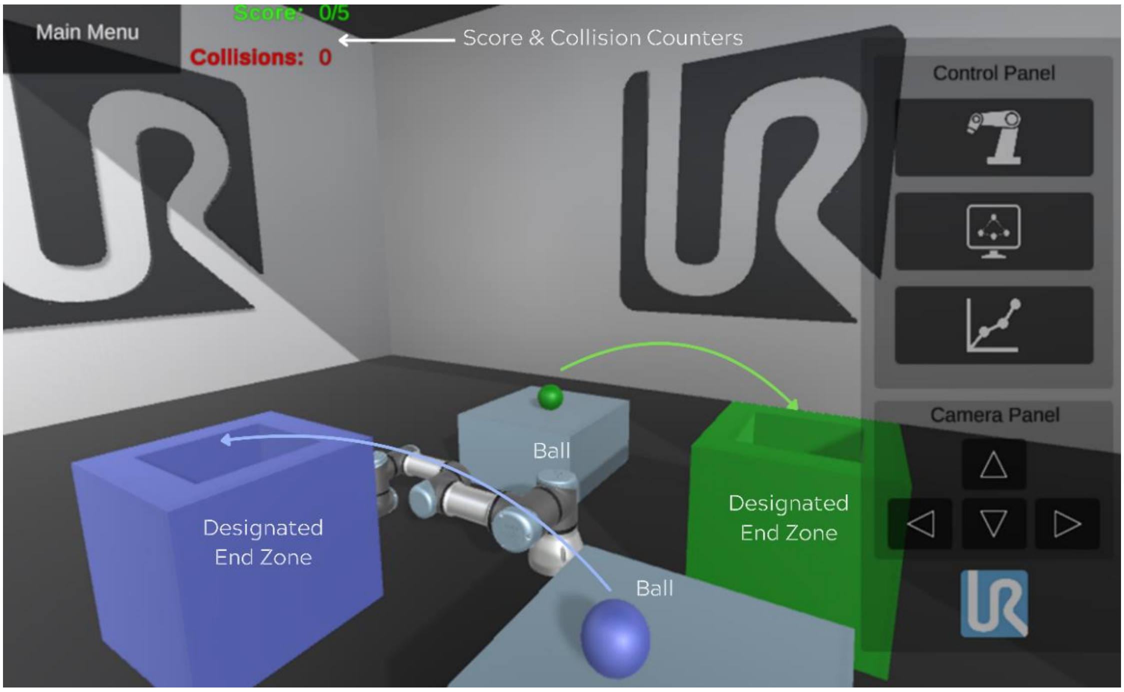

Figure 3: Level 4

Level 5 restricts the path that can be taken from the object to the zone. Students are forced to create precise movements and utilize multiple joints of the robot. This level reinforces lessons learned from previous level with the addition of accurate and precise movement. A maximum collision can be easily set to fail/restart the level to provide a challenging experience. Level 6 is identical to level 5 but again with increased score requirements to force coding through repetition. Level 5 is shown in Figure 4: Level 5.

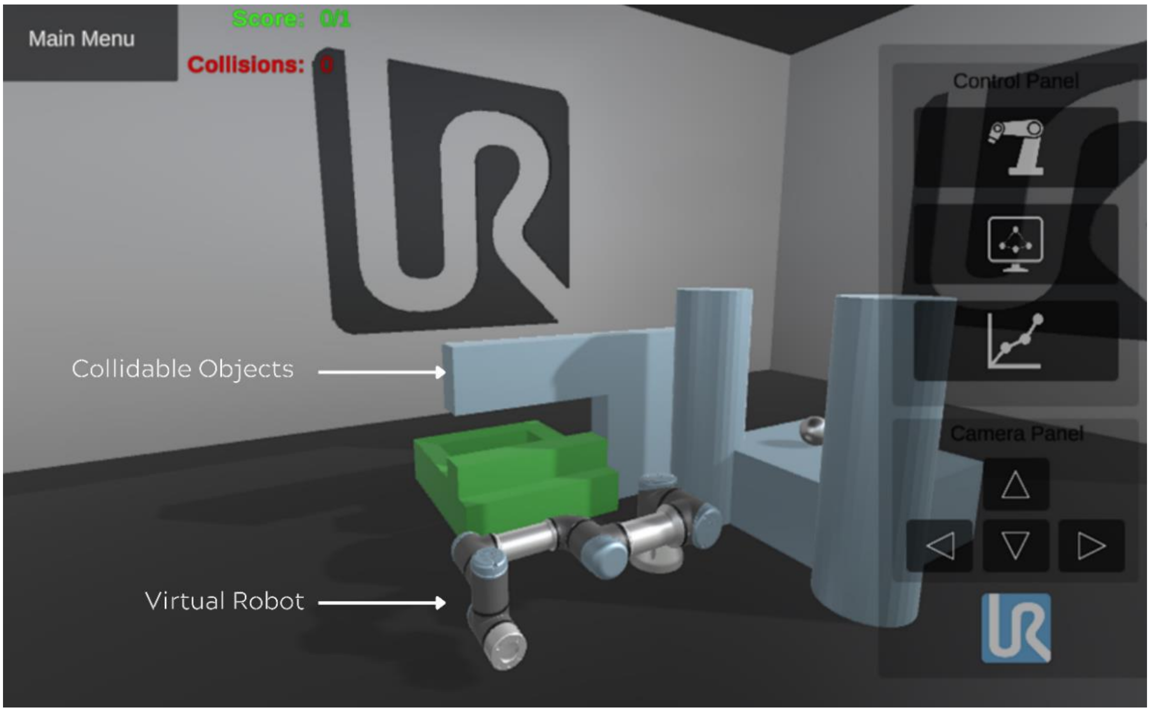

Figure 4: Level 5

The rationale for each level being performed twice-once with a required score of one, and then again with an increased required score- is to first provide students an opportunity to manually control the robot and perform the task. This will allow them to formulate a path and reference to create and optimize their coding for repetition.

3.4. Application Architecture and Logic

In developing the learning app, it was crucial to implement coding logic that closely mirrors real-world tasks. However, given the scope of the project and its focus on delivering a proof of concept, certain design decisions were made to prioritize functionality and ensure the creation of a 'Version 0.1'. These decisions helped balance the app's educational objectives with the technical limitations, providing a foundation for future iterations and enhancements.

3.4.1. Network Communication and User Navigation

As described above in Technical System Overview, the URSim software sends co-ordinates and positional data to the learning application. This is done with Transmission Control Protocol. This protocol enables programs and computing devices to adhere to a communications standard to exchange messages over a network. This protocol allows the positional data to be sent from the laptop to the android tablet when connected to the same network. This connection was achieved by Mr. Roman Parak.

The ‘Main Menu’ screen utilizes simple buttons that load ‘Scenes’ in the application, each level was assigned a scene number and when the corresponding button was pressed, it would run that scene/level. Identical logic exists for the ‘Main Menu’ button that returns the user to the main menu when a level is loaded. This is the navigation system used for the learning app.

3.4.2. Object Interaction: Pick-and-Place Mechanism

A ball was chosen as the object that was to be moved from a starting point to a designated zone to complete the simple pick-and-place tasks required to complete the levels. The ball was assigned the “PICKUPABLE” tag using the Unity tag system that allows a developer to easily assign attributes to objects by allowing attributes to be linked to the tag. A small invisible object with a ‘sphere collider’ was placed at the front of the virtual robot’s TCP, this object was named the ‘MagnetPoint’. This collider would be the position of the ball if it was picked up. It was also the zone where the robot would pick up a ball if the two colliders intersected.

A script was created that is activated when the MagnetPoint's collider, collides with another object's collider. The script firstly checks whether something is already picked up, if something was already picked up, the robot could not pick up a second object. The script secondly checks whether the object has the tag "PICKUPABLE", if the object has the tag, it will assign the object's positional data to that of the MagnetPoint and turn off the object's physics and kinematics, rendering it a part of the robot and will follow along while remaining in the MagnetPoint position. If it does not possess the required tag, no interaction will take place. A visual illustration is presented in Figure 5: Object Interaction: Pick Up Mechanic.

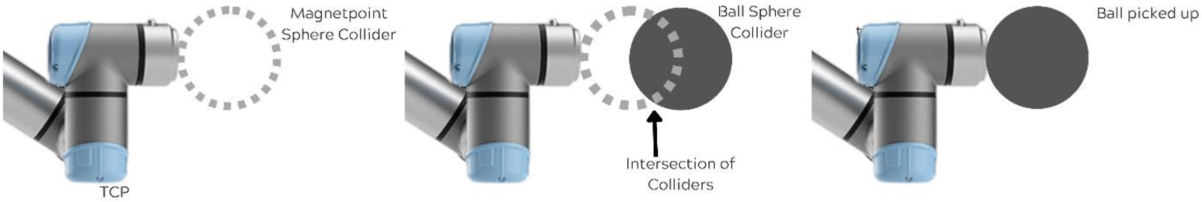

Figure 5: Object Interaction: Pick Up Mechanic

3.4.3. Target Area and Respawn Mechanics

The designated zone where in the user must drop the ball is a box with an open hole. The box features two invisible zones with rectangular colliders. One zone is slightly above the open hole in the centre of the box, this is the release point or 'TARGET AREA', the second zone is located at the bottom of the box above the floor, this is the 'RespawnTriggerArea'. The TARGET AREA has the tag 'GOALZONE', in the same script as mentioned above to pick up the ball, the MagnetPoint checks whether it has entered the collider of an object with the tag 'GOALZONE', if this is true it will run a function 'releaseball', which releases the object and turns on the object's physics and kinematics properties. The function also enables the MagnetPoint's ability to pick up PICKUPABLE objects. A visual illustration is presented in Figure 6: Target Area and Respawn Mechanics.

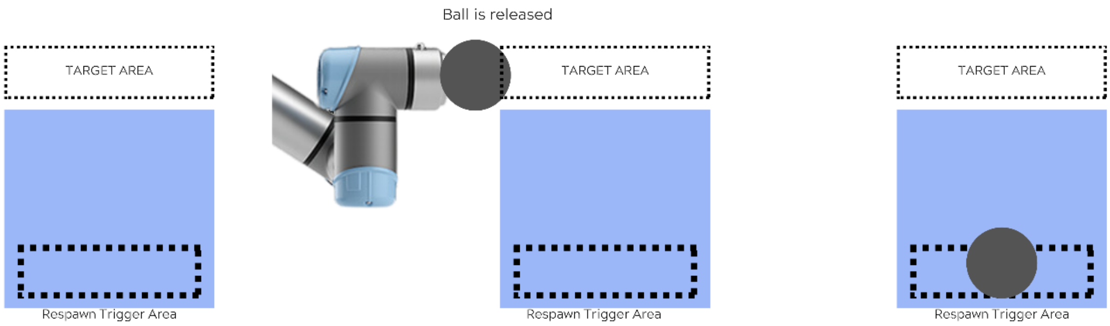

Figure 6: Target Area and Respawn Mechanics

The ball object features a script that detects whether the ball's collider has collided with another object's collider, if this object has the tag 'RespawnTriggerZone', it will release the MagnetPoint to allow it to pick up a new ball. In the 'Scene Editor', if there are multiple balls as in Level 3 and Level 4, a property is assigned to each ball to define which RespawnTriggerZone is associated with each respective ball. If the wrong ball is in the wrong RespawnTriggerZone, it will call the 'RespawnBall' function, which will return the ball to its initial starting point or spawn point and set the ball's velocity and angular velocity to zero, to avoid the ball falling off the table where it spawns. If the correct ball is in the correct RespawnTriggerZone, it will call the 'IncreaseScore' function. This function is defined in the 'ScoreCard' script.

3.4.4. Scoring System and Collision Detection

The IncreaseScore function increases the counter for the number of balls in the correct zones, each level has a property that defines the number of balls in the level. The function compares the number of balls in the level to the number of balls in the correct zone, if they are equal, it will update the score by one and reset the counter for balls in the correct zone. The function then loops through all objects with the PICKUPABLE tag and executes the RespawnBall function.

A 'UpdateScoreDisplay' function updates the text that displays the current score. The function also compares the current score to the level's target score, if this target score has been reached it will display: "Level Completed!". The function also updates the number of collision as they occur. All objects that are not associated with the PICKUPABLE tag, are given the 'Collidable' tag. The platforms on which the balls spawn is given this tag along with any other obstacles. The entire robot model is given colliders to fit each respective joint, on the main object a script checks whether any of these joint colliders have collided with another object. If these objects have the Collidable tag, the script will call the IncreaseScore function to update the number of collisions. It should be noted that when the ball is attached to the MagnetPoint, it will also count as a collision, to ensure that users do not bump the object as in reality, a part should not be dragged along a table or accidentally collide with other objects/obstacles.

The above section describes the technical overview of the virtual environment, and the objects created within this environment along with their interactions and functionality. The objects, scripts and properties were created to allow for future iterations of the learning app with new levels to be created with ease and minimal effort.

3.5. Educational Tools and Feedback

The development of the educational tool for this project was guided by various learning frameworks and assessment models, particularly those covered in the Lens of Learning literature review. These frameworks ensure that the virtual environments and tasks created for students align with sound educational practices, provide meaningful learning experiences, and gather comprehensive feedback on the learning material's effectiveness. Tait's Model, the ADDIE Model, and Revised Bloom's Taxonomy were integral to structuring the learning experience within the virtual environment.

3.5.1. Tait's Model

Tait's Model, with its emphasis on gradual scaffolding, allowed for an incremental increase in difficulty as students progressed through each stage. This structured approach provides students with manageable learning steps, helping them build on prior knowledge and gain confidence in their ability to control industrial robots. The adaptability and scalability of this model also proved valuable, as it allowed the virtual environment to be easily adjusted and replicated for different learning scenarios.

3.5.2. ADDIE Model

The ADDIE Model (Analysis, Design, Development, Implementation, and Evaluation) was used to systematically design the virtual environments and learning tasks. In the Analysis phase, the needs of students, specifically those learning to program and control industrial robots, were examined, including the challenges they face when working with physical robots. During the Design phase, the stages in the virtual environment were structured based on incremental task complexity, aiming to engage students while gradually introducing more advanced concepts. The Development phase involved creating the virtual environment and aligning each stage with educational objectives. In the Implementation phase, the tool was introduced to a pilot group of students, as will be outlined in the pilot study section. Finally, the Evaluation phase analysed feedback and performance data to be used for refining the stages and tasks to ensure optimal learning outcomes.

3.5.3. Revised Bloom's Taxonomy

RBT was applied to shape the cognitive objectives of each task within the virtual environment. The taxonomy was used to ensure that tasks engaged students across multiple levels of cognition, from basic understanding and application to higher-order skills like analysis, evaluation, and creation. For example, initial stages involved recalling fundamental robot control concepts (such as basic movements), while later stages required more complex decision-making and problem-solving (such as optimizing robot movements). This progression ensured that students not only mastered technical skills but also developed critical thinking and problem-solving abilities relevant to industrial robot operation.

3.5.4. Feedback

In designing the feedback questionnaires, NASA TLX was employed to measure the cognitive workload of students as they completed tasks, ensuring that the difficulty of each stage was balanced to challenge students without overwhelming them. Additionally, the PANAS questionnaire captured the students' emotional responses throughout their learning journey, allowing for a well-rounded evaluation of the educational tool's impact on both cognitive and emotional levels. Additional general feedback questions were added to the modified TLX and PANAS frameworks to provide a holistic feedback system.

3.6. Pilot Study

The pilot study is a crucial step in testing and refining the virtual environment and its learning material before broader implementation. It involved a small group of participants, comprising of eight subjects, to assess the effectiveness of the virtual environment in teaching cobot control and programming skills. This preliminary phase is designed to identify any potential issues, gather initial feedback, and refine both the virtual environment and the assessment tools.

The virtual environment for this study was developed based on Tait's Model, the ADDIE Model, and Revised Bloom's Taxonomy. These models ensure that the learning experience is structured in a way that allows for scalable and adaptable progression through stages of increasing complexity. The educational scaffolding provided by these models supports gradual skill acquisition, ensuring that the students build their competencies step by step. Feedback was collected through questionnaires, designed using NASA TLX for workload assessment, Revised Bloom's Taxonomy to evaluate the cognitive impact of the activities, and PANAS to measure the emotional responses of participants throughout the learning experience.

3.6.1. Procedure

1. Introduction: The pilot study began with a brief introduction explaining the significance of industrial robots and cobots in the context of modern industry. Participants received an overview of robot control and programming techniques, simulating a lecture or tutorial setting.

2. Familiarization with Freeplay: Before attempting the structured stages, participants were encouraged to explore the Freeplay mode to familiarize themselves with the controls and the virtual robot. This mode provides an opportunity to understand how to operate the robot without the pressure of completing specific tasks.

3. Attempting the Stages: After the familiarization phase, participants progressed through the designed stages in ascending order. Each stage increases in difficulty, reflecting the incremental scaffolding approach used in the environment's design.

4. Feedback Collection: At the conclusion of the session, participants completed feedback questionnaires. These captured the subjective workload (NASA TLX), cognitive difficulty (Revised Bloom's Taxonomy), and emotional impact (PANAS). This feedback is invaluable for assessing the pilot study's effectiveness and guiding further iterations of the virtual environment.

The pilot study is an essential phase in ensuring that the final virtual environment and its associated educational tools are both effective and user-friendly. By running this preliminary test, the objective is to optimize the overall design and assessment strategy for a broader implementation.

3.7. Data Collection and Analysis

The data collection and analysis process in this project focuses on gathering qualitative feedback to evaluate the educational tool's effectiveness in teaching students to use and program cobots. This was achieved through feedback questionnaires and observational data, both of which offered insights into students' experiences, cognitive workload, and emotional responses during their interaction with the virtual environment.

3.7.1. Data Collection Methods

Data was collected primarily through two methods: feedback questionnaires and observations during the pilot study.

- <u>Feedback Questionnaires:</u> Upon completing the stages, students filled out questionnaires designed using NASA TLX, Revised Bloom's Taxonomy, and PANAS. The NASA TLX questions assessed the students' cognitive workload, focusing on factors such as mental demand, effort, and frustration levels. Revised Bloom's Taxonomy questions asked students to reflect on their cognitive experience during the tasks—such as their ability to understand, apply, and analyse new information. PANAS measured the emotional responses to the tasks, capturing both positive and negative feelings. The responses were automatically captured to a google sheet for storage and analysis.

- <u>Observational Data:</u> Observations were made during the pilot study, focusing on students' interactions with the virtual environment, how they approach tasks, and their general engagement. This qualitative data provided further context to the feedback questionnaires, giving a more detailed understanding of the challenges and successes students encountered.

Beyond these structured assessments, students were also asked for general feedback on the virtual environment, including whether they encountered any bugs, how they perceived the quality of the environment, and suggestions for potential improvements. This assisted in evaluating the technical and user experience aspects of the virtual environment, contributing to its refinement.

3.7.2. Data Analysis Methods

The qualitative data collected was analysed to understand the educational tool's impact on student learning and the overall user experience:

Qualitative analysis was executed with the feedback from the questionnaires. The feedback was analysed thematically to identify patterns in student responses. This analysis highlights areas where students faced cognitive overload, struggled with tasks, or felt successful in their learning. Emotional responses from the PANAS-based questions were categorized into positive and negative experiences, providing insights into students' engagement and enjoyment. Cognitive reflections based on Revised Bloom's Taxonomy assisted in assessing the depth of learning, highlighting areas where students successfully advanced in understanding and application.

Additionally, responses related to the quality of the virtual environment—such as feedback on bugs, visual quality, and potential improvements—were reviewed to identify areas where the technical aspects of the environment can be optimized for smoother user experience.
The observational data was reviewed for common behaviours and notable instances where students either engaged deeply or encountered difficulties. This was cross-referenced with the questionnaire data to provide a fuller picture of students' interactions with the virtual environment.

3.7.3. Triangulation

By combining feedback from the questionnaires with observational data, a comprehensive understanding of the educational tool's strengths, areas for improvement, and the virtual environment's technical quality was developed. The combination of cognitive load, emotional response, and students' reflections on the stages guide refinements of the tool.

3.7.4. Refinement

Based on the analysis, adjustments were made to ensure the virtual environment provides an appropriate balance of challenge and engagement. If certain stages consistently result in cognitive overload, negative feedback, or technical issues such as bugs, they should be redesigned to better align with the learning objectives. This iterative process ensures that the tool remains effective, user-friendly, and technically sound, offering a supportive and educational experience.

3.8. Conclusion of Methodology

The methodology section has outlined a comprehensive and structured approach to the development, testing, and evaluation of a virtual educational tool designed to teach students how to program and operate cobots. By building upon existing tools such as Roman Parak's Unity model and integrating URSim within a virtual environment managed by VMware, the project creates a realistic and scalable learning platform for students. The use of proven educational frameworks, including Tait's Model, the ADDIE Model, and Revised Bloom's Taxonomy, ensures that the virtual stages are instructional and scaffolded to facilitate progressive skill development.

Additionally, the feedback mechanisms incorporating NASA TLX and PANAS help to assess cognitive workload and emotional responses, ensuring a balanced and effective learning experience. The pilot study further refines the tool, offering a chance to gather preliminary data and make necessary adjustments before broader implementation.

By following this iterative and feedback-driven process, the project aims to create a robust educational tool that enhances student understanding and proficiency in cobot programming, bridging the gap between theoretical knowledge and practical skills. The methodology provides a clear roadmap for achieving the project’s objectives and ensuring the app is both functional and pedagogically sound.

Chapter 4 - Results

This section discusses the results of the feedback obtained from the pilot study. The feedback questionnaire was created to assess:

- the effectiveness of learning outcomes,
- the design of the learning methods,
- the perceived workload experienced,
- emotional experience,
- learning app feedback,
- learning app improvement areas,
- and overall experience.

Additionally, observational data was taken to supplement the questionnaires by observing the user's methods, feelings, and usage of the app.

An initial test was run without the use of instructional content with a single subject to obtain information on what content would be needed for the pilot study and the estimated time needed per student. The selected structure was that of a small introduction presentation that resembled a short lecture which then was then followed by a tutorial like setting where a student completes a series of tasks with assistance, if needed, from a facilitator. A photo of a participant performing the pilot study is shown in Figure 7: Pilot Study Procedure.

Figure 7: Pilot Study Procedure

The introduction presentation was conducted using a PowerPoint presentation where the facilitator gave a short introduction to the purpose of the project, a brief background of ICR's and their uses, simple instructions on how to use the URSim software and the learning app on the tablet. After the completion of the task, the participant was asked to complete the questionnaire. Participants were told that their performance would not be measured and were not given details on the level of completion required for the tasks, it would be their own discretion on the number of collisions acceptable and when they were finished with the task at hand. The pilot study featured eight (8) participants excluding the initial test subject. All participants were Engineering students at Stellenbosch University, ranging from 1st year to Master students.

4.1. Feedback from Questionnaires

This subsection reviews quantitative and qualitative data obtained through the feedback questionnaires; each page of the questionnaire reviewed different aspects that relate to the educational models selected. Majority of the questions utilize the Likert scale where participants use a scale from 1-5 to quantify their response, these questions feature an optional "Reasoning or additional comments" follow-up question that allows the participant to provide context to their answers if they wanted to. Qualitative questions were asked where students could answer in either short or longer text format. The full questionnaire can be viewed in the Appendices along with all results.

4.1.1. Interest Levels Pre- and Post-Session

Participants were asked to score their interest in ICRs before and after the session, on a Likert scale from 1 (Not at all) to 5 (Extreme). These questions assess the impact of the learning app and pilot study activity on the participants' interest in ICRs. Figure 8: Participants' Interest Before and After Pilot Study showcases the results of these questions.

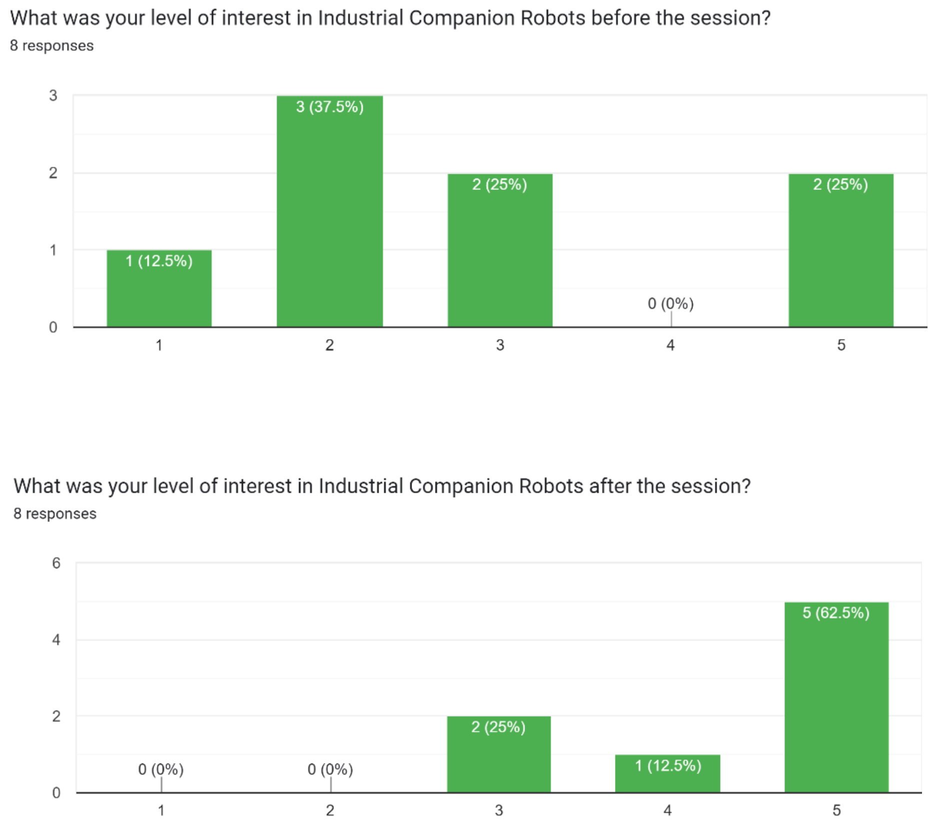

Figure 8: Participants' Interest Before and After Pilot Study

The first chart, interest before the session, participants' interest was spread throughout although predominantly low to moderate. Most of the participants selected 2/5 as their interest level (37.5%) before the session. The second chart, interest level after the session, shows a significant shift. After the session 62.5% of participants rated their interest the highest level (5/5), a significant increase from the original 25% before the session. This data suggests that the pilot study was able to effectively increase the participants enthusiasm and engagement with the content material and highlights the potential of the learning app to positively influence student engagement with ICR's.

4.1.2. Revised Bloom's Taxonomy

The first section of the questionnaire was created to assess the cognitive objectives of the tasks within the virtual environment. This was done using the RBT tiered model. The questionnaire questions were structured to determine the level of cognitive ability the participants felt they achieved through the session. The participants all (100%) indicated a 5/5 ability to apply the knowledge obtained to perform the tasks in the learning app, which corresponds to the first tier of the RBT model. This confirms the basic levels of remembering and understanding the material is effectively satisfied.

7 out of 8 (87.5%) participants scored a 5/5 when asked whether the learning app encouraged the breaking down and analysis of the robot's movement to solve the tasks. The remaining student scored a 4/5 in this question. This indicates that the students were able to effectively apply and analyse problems presented in the learning app. Notable additional comments include that "Certain tasks required different movements encouraging the user to fully understand the robot movement in order to complete the task.". This confirms the tiers that states that effective learning should include applying the knowledge obtained through analysing complex problems to formulate a solution.

The 5th tier of evaluating refers to a student's ability to evaluate their solution to a problem. The corresponding question asked participants how well the learning app allowed them to evaluate the effectiveness of your task performance and programming solutions. 7 out of 8 participants (87.5%) scored a 5/5, and 1 participant scored a 4/5. Additional comments included that the collision counter was effective in this aspect and the visualization of the learning app allowed them this opportunity.

The final tier of cognitive learning is the Creating tier. The final question in the section asked participants to what extent did the learning app enable them to develop new programming strategies during the tasks. The scale for the answers were 1 – Not at all to 5 – Extremely as shown in Figure 9: Participants Responses to Creating Tier Question.

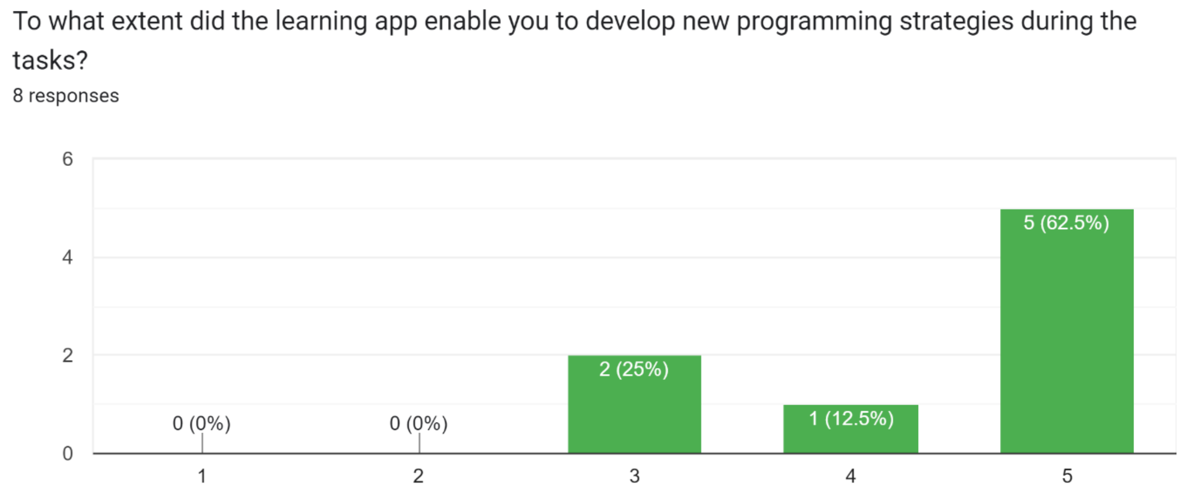

Figure 9:Participants Responses to Creating Tier Question.

It is evident that a majority (62.5%) felt that they were able to develop new strategies that they used to complete the tasks. Some more experienced participants felt that more features could enhance the ability to develop new strategies such as the ability to view the degrees and co-ordinates of the robot in the learning app, as this information is only available on the robot controller.

The overall feedback suggests that the learning app is effective in teaching students core principles of ICR programming across all tiers of cognitive development as described by the RBT model. In later versions of the learning app, further levels and techniques can be introduced to allow students the ability to develop more complex programming strategies to better satisfy the final tier of creating.

4.1.3. ADDIE Model

The feedback questionnaire was used to Implement and Evaluate the effectiveness of the learning app design. The last two steps of the ADDIE model used to design the virtual environment.

The first question of this section asks the participants whether the layout of the app was easy to navigate and understand, on a scale of 1 – Very easy to 5 – Very difficult. The results are shown in Figure 10: Participants' Responses to App Layout and Navigation.

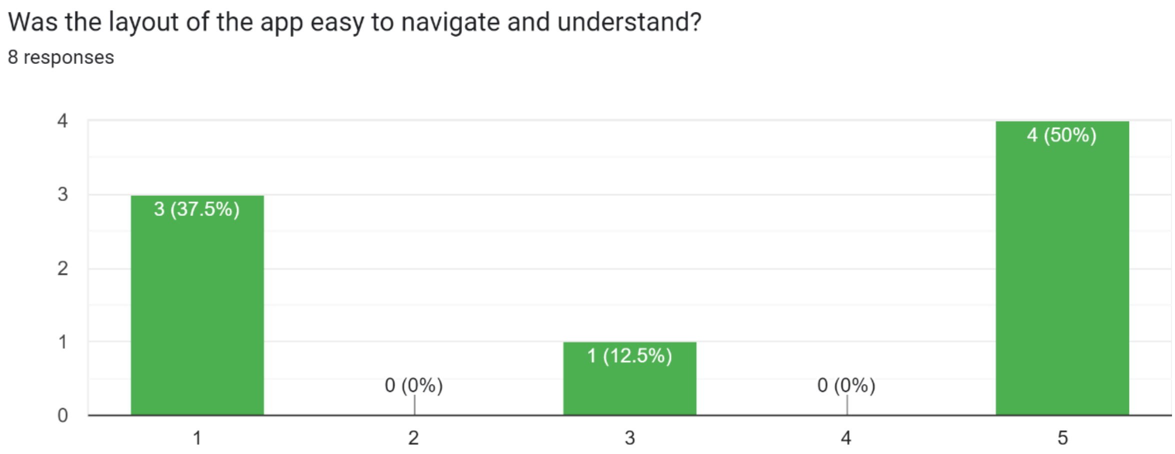

Figure 10: Participants' responses to app layout and navigation.

The results are divided although the facilitator of the pilot study believes that this question has been influenced by the URSim robot controller application as it features an un-intuitive design with many tabs and layers. As this is the robot controller for not only the simulation but also the actual controller for Universal Robots cobots, this cannot be changed without jeopardizing the learning materials effectiveness to industry simulation.

Most found the layout and navigation to be very difficult to use and understand (50%). Although almost half of the participants found it very easy to use (37.5% or 3/8 participants). Due to this large divide in scores and observations made by the facilitator during the pilot study, it was concluded that the learning app navigation is sufficient although could benefit from refinement.

The second question of the section that evaluates the pilot study according to the ADDIE model asked participants whether the instructions were clear and helpful for completing the tasks in the learning app. The results can be seen in Figure 11: Participants' Responses to Instructions Effectiveness.

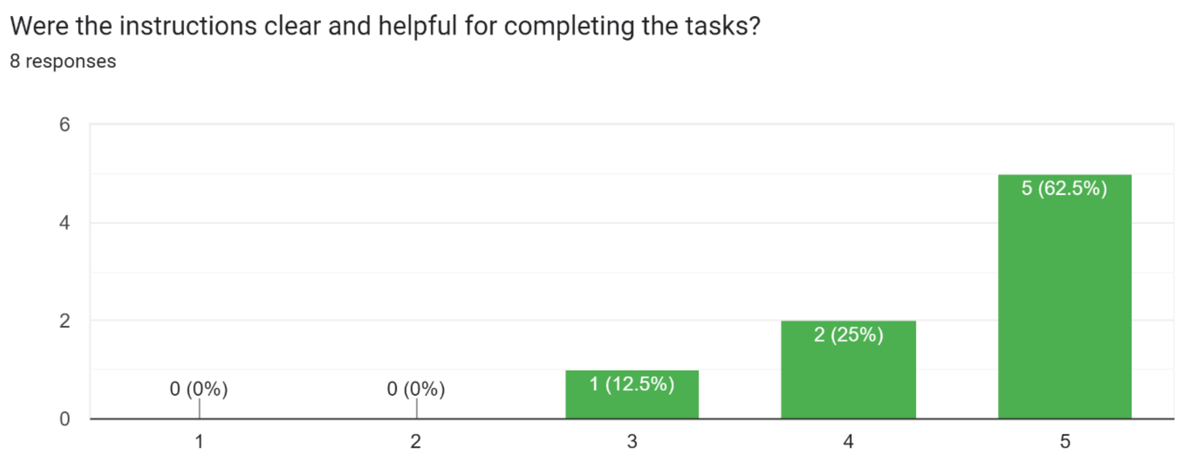

Figure 11: Participants' Responses to Instructions Effectiveness.

Most (62.5%) participants found the instructions to be Extremely helpful (5/5), although 37.5% of participants believed there was room for improvement although the current state was acceptable. Further creation of the instructions around the learning app such as formal lectures and tutorials can supplement the required gap alongside additional functionality of the app with a help or hint button to assist users in each level.

The remaining questions in this section provided feedback that the learning app was able to keep participants engaged (75% scored the maximum of 5/5), and that the feedback tools effectively helped them monitor their learning (87% scored the maximum of 5/5). This section revealed that with small refinements and 'quality of life' features the learning app is a designed and developed adequately to function as a learning tool. Supporting instructions and learning material can also be refined when presented in a more formal lecture environment.

4.1.4. NASA TLX

This section utilizes the NASA TLX workload exertion assessment tool with slight modifications to evaluate the perceived workload experienced by pilot study participants. The participants were asked to rate the mental demand of the tasks, the effort they had to exert to complete the tasks, whether any and to which level of frustration the experienced and how satisfied they were with their overall performance. Figures 13 to 15 showcases the result

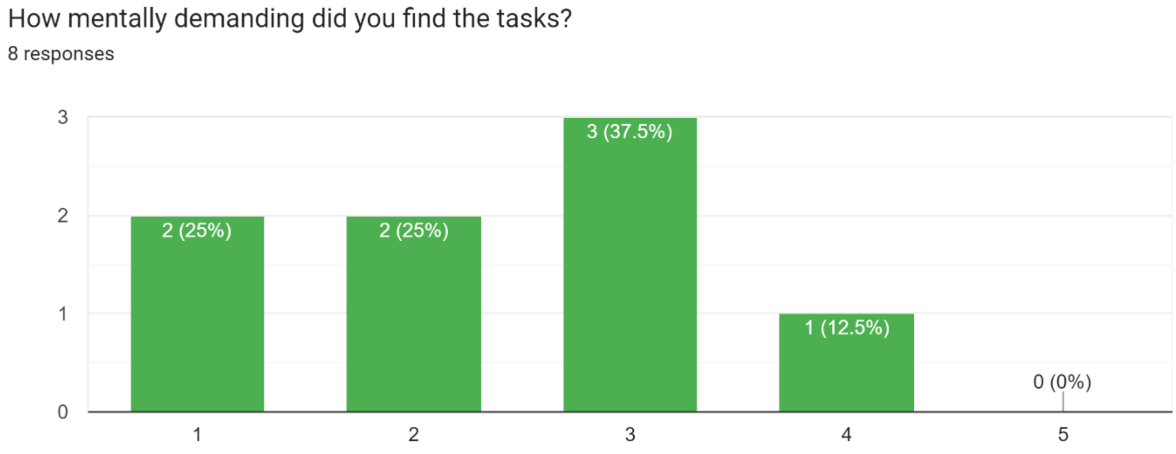

Figure 12: Participants' Responses for Mental Demand

Figure 12: Participants' Responses for Mental Demand shows the mental demand experienced by participants with the scale 1 – Very low to 5 – Very high. An almost equal distribution exists, this showcases that the mental demand is unique to each participant and various factors influenced their chosen score, these factors include but are not limited to level of perfection when completing the levels (completing all levels with 0 collisions in final solution) and previous experience working with 3D vector spaces (such as CAD). It should be noted that no participant indicated 5/5 (very high) mental demand, this can be interpreted that although moderate-high mental demand was required from 50% of participants, none felt it was extremely demanding. This shows that although the levels were challenging, none felt it to be

overly mentally demanding.

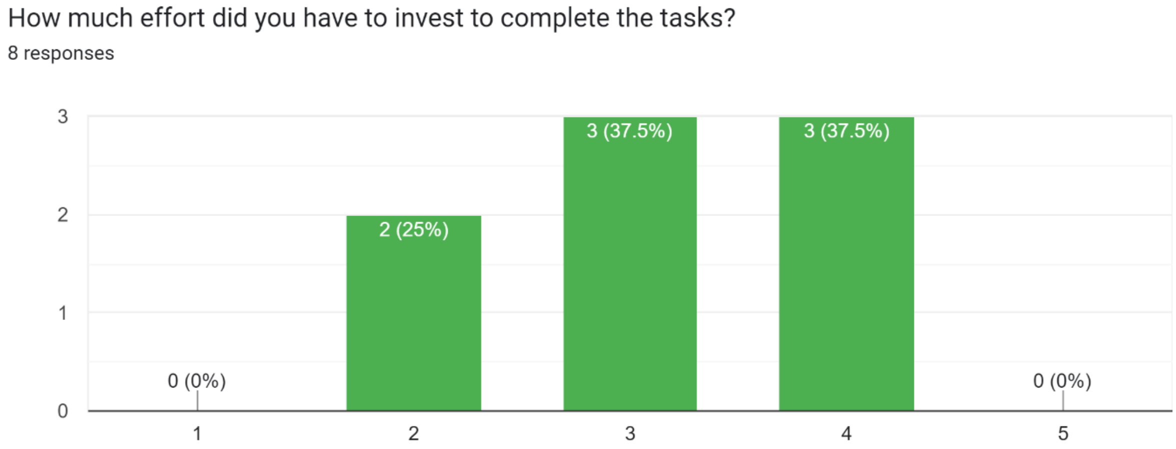

Figure 13: Participants' Responses for Effort Required

Figure 13: Participants' Responses for Effort Required illustrates the distribution of perceived effort exerted by participants to complete the tasks. It is centrally distributed around the moderate effort level (3/5). This showcases that the tasks in the learning app are challenging although do not overwhelm participants.

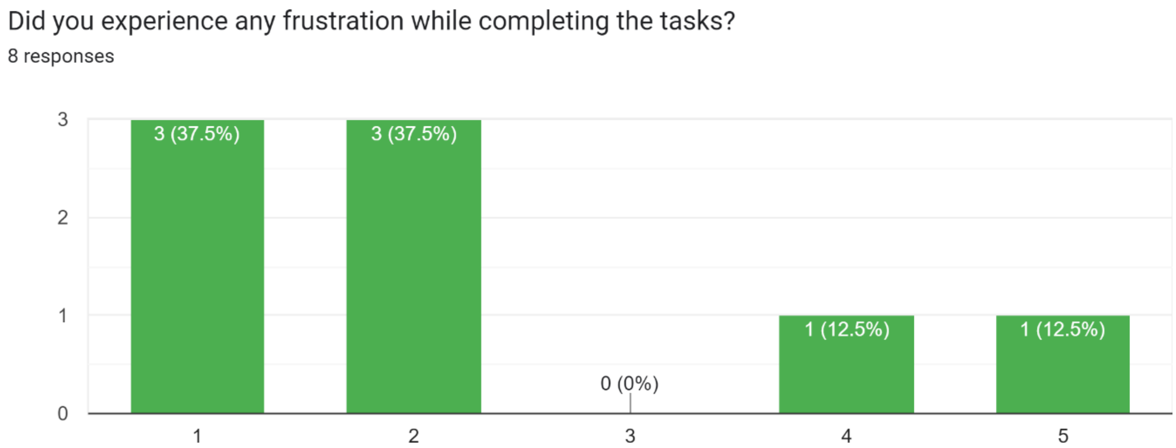

Figure 14: Participants' Responses for Frustration Experienced

Majority of participants experienced none to little frustration while completing the tasks as shown in Figure 14: Participants' Responses for Frustration Experienced. Two participants experienced high to extreme frustration while completing the tasks. The additional comments reveal that bugs experienced while using the app contributed to these frustrations. One user for example had multiple disconnections between the tablet and the URSim software.

The last question of this section asked participants to indicate the level of satisfaction they felt in relation to their overall performance. The facilitator explained in the introduction to each participant that their performance would not be evaluated or recorded. The only indicator of performance was the completion of the stages by achieving the required amount of score for each stage. Although there was no limit to collisions and participants could decide for themselves what was sufficient and when to stop. All participants did finish all stages, and the majority did complete each stage with a working robot path that featured zero collisions. While other participants' robot paths featured one or two collisions in their final path although deemed this to be acceptable to their standard.

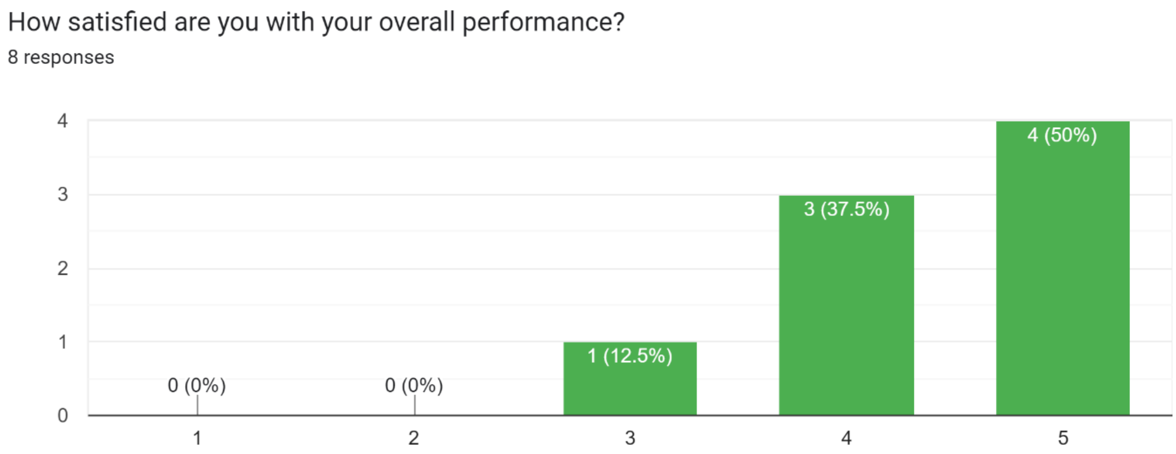

Figure 15: Participants' Responses for Overall Performance

Figure 15: Participants' Responses for Overall Performance indicate that 50% of the participants felt the maximum (5/5) satisfaction with their overall performance and 37.5% of participants felt that their performance was highly satisfactory (4/5). One participant scored the midpoint average value of 3/5. The facilitator noted that all participants had a reaction to the collision counter that influenced their self-perceived performance. Thus, the high scores indicate that all participants felt they had accomplished the tasks although the number of collisions during their tasks influenced their sense of performance.

4.1.5. PANAS

The PANAS questionnaire was slightly modified to evaluate the emotional experience of participants during the pilot study in this section of the questionnaire. Different emotions were to be score on a scale of 1- Not at all to 5- Extreme. The emotions in questions ranged from positive – negative emotions and the participants overall emotional experience.

The level of interest and level of excitement were featured as the first positive emotions and participants results were similar. 7 of the 8 (87.5%) participants rated these emotions as either 4/5 or 5/5 indicating high-extreme interest and excitement while using the learning app. 1 participant rated both these aspects as a low-level (2/5). The results are shown in Figure 16: Level of Interest and Excitement Participants Experienced.

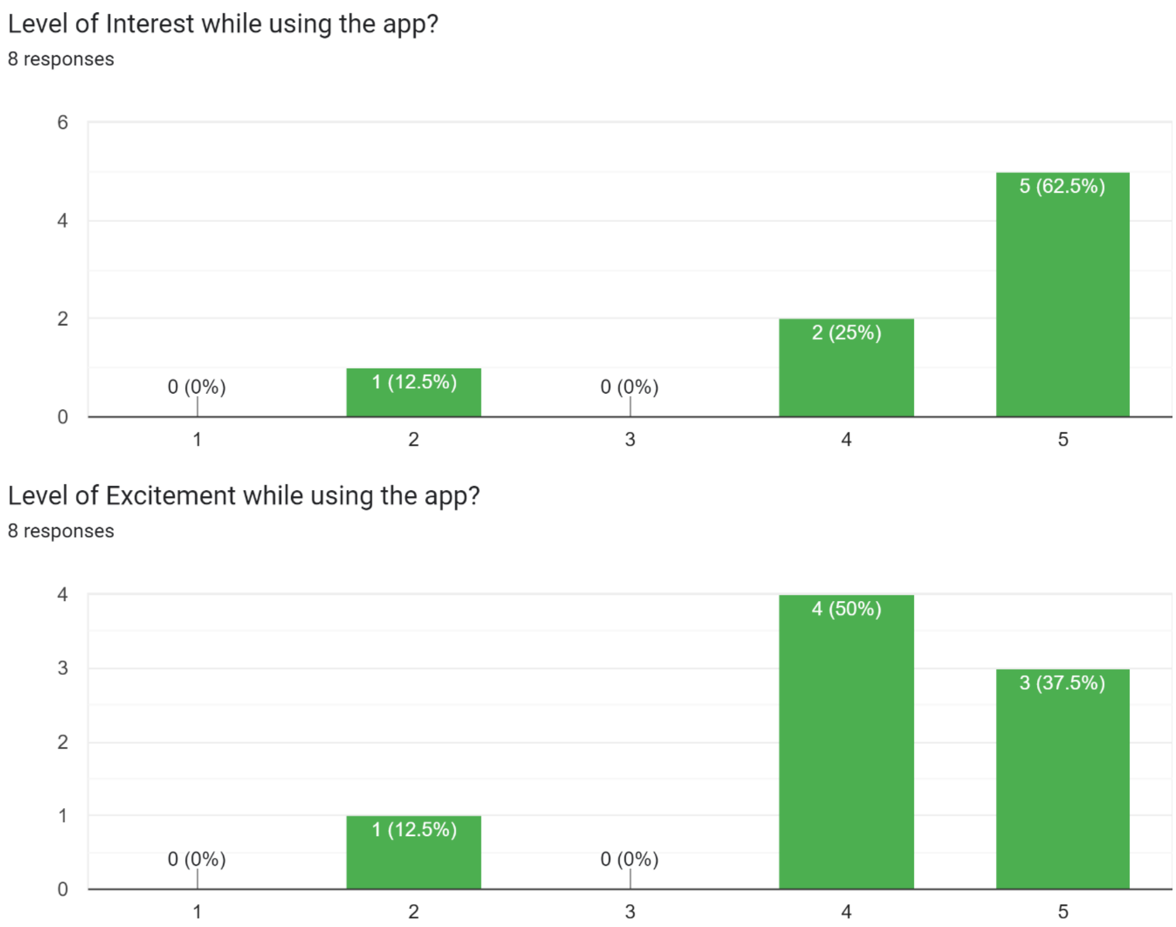

Figure 16: Level of Interest and Excitement Participants Experienced

The feedback suggests that the learning app was successfully able to create interest and excitement while completing the tasks for the vast majority of participants. This is essential as learning is more effective if students are excited and interested afterwards, creating a lasting impression of the learning material. The one student that indicated a low score noted that large movements of the robot took some time when controlling the robot manually, and that they could "chat or become focused on something else". The speed of the robot during manual movement is limited in the native URSim controller for safety considerations although the stages did feature multiple of these large movements for the most obvious solutions.

Participants responses for the level of inspiration and level of determination featured identical results as seen in Figure 17: Participants Level of Inspiration and Determination. The results feature a normal distribution that skews to the left towards more positive results. The median answer of high-level (4/5) featured 50% of the participants.

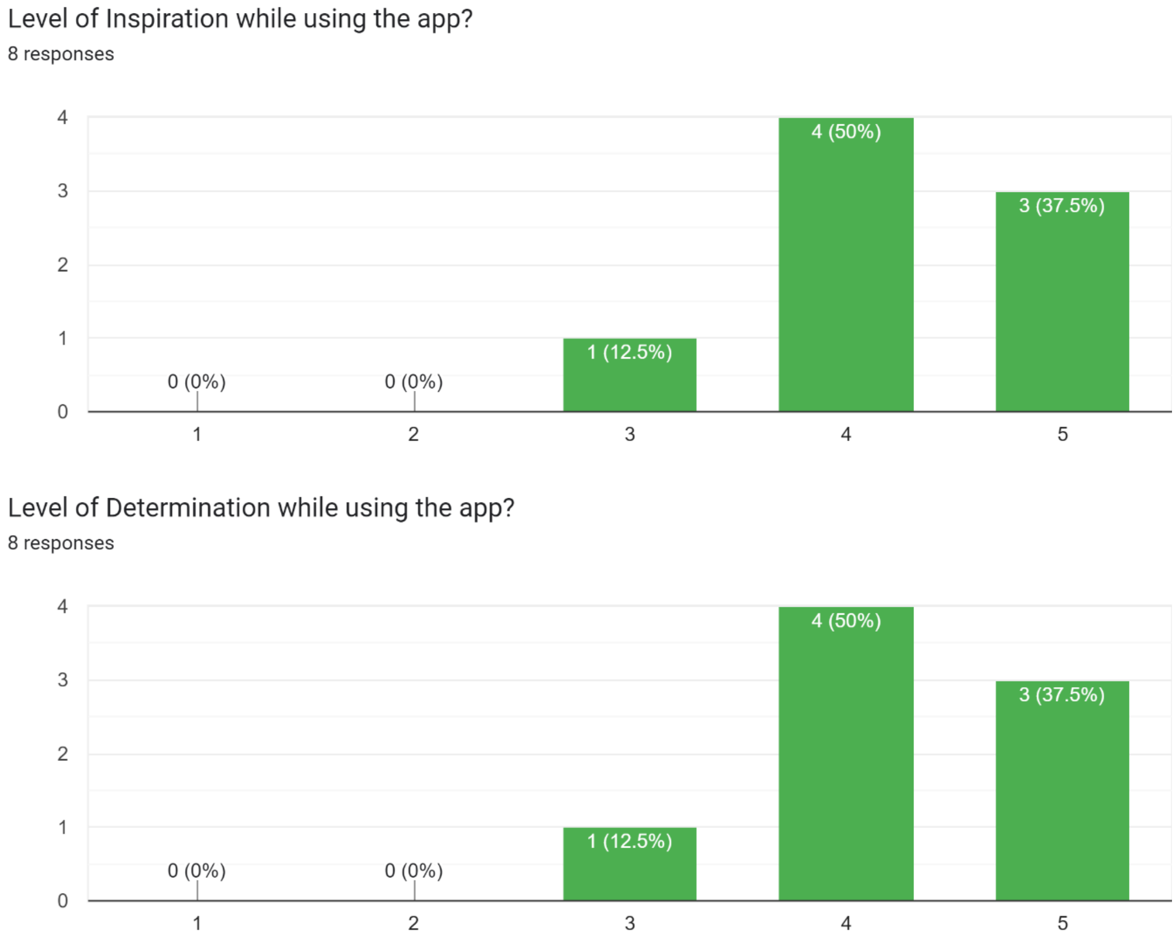

Figure 17: Participants Level of Inspiration and Determination.

These emotions are interrelated as the more inspired a participant felt, the more determined they are to complete the tasks presented. These results indicate that the learning app was successful in inspiring participants to complete the tasks.

The final positive emotion measured was the level of attentiveness while using the app. The results depicted in Figure 18: Participants' Level of Attentiveness, are again highly positive as 5 out of 8 (62.5%) of participants indicated the maximum score of extreme attentiveness (5/5).

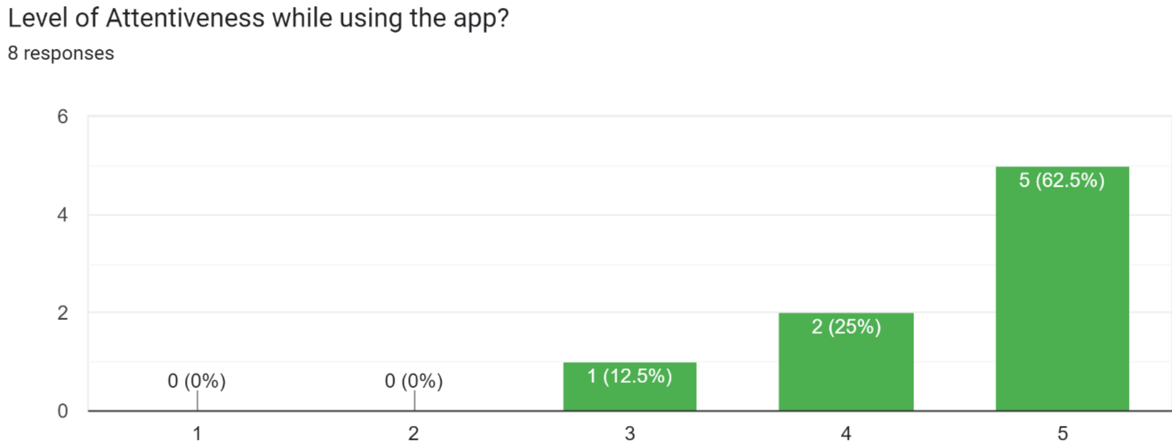

Figure 18: Participants' Level of Attentiveness

This confirms an important objective of any educational tool of retaining attention during the learning process is essential. The results indicate that the learning app effectively satisfied this objective.

The negative emotions experienced were assessed using the same Likert scale of 1-5. Negative motions assessed featured the level of Nervousness, Upset, Distress, Frustration, Confusion, and Fear participants experienced during the pilot study. The emotions received similar feedback as most participants indicated a 'Not at all' (1/5) response for each of these negative emotions. The graphs for each of these results are attached in the appendix. Additional comments relating to the negative emotions revealed that participants felt most of these emotions during the start of the learning app while learning to use the robot controller and the identification of each joint. These comments are further discussed in the key-findings and recommended improvements sections.

The concluding question of the PANAS section asked participants to rate their overall emotional experience with the learning app on a Likert scale of 1- very negative to 5- very positive. The results are shown in Figure 19: Participants Overall Emotional Experience.

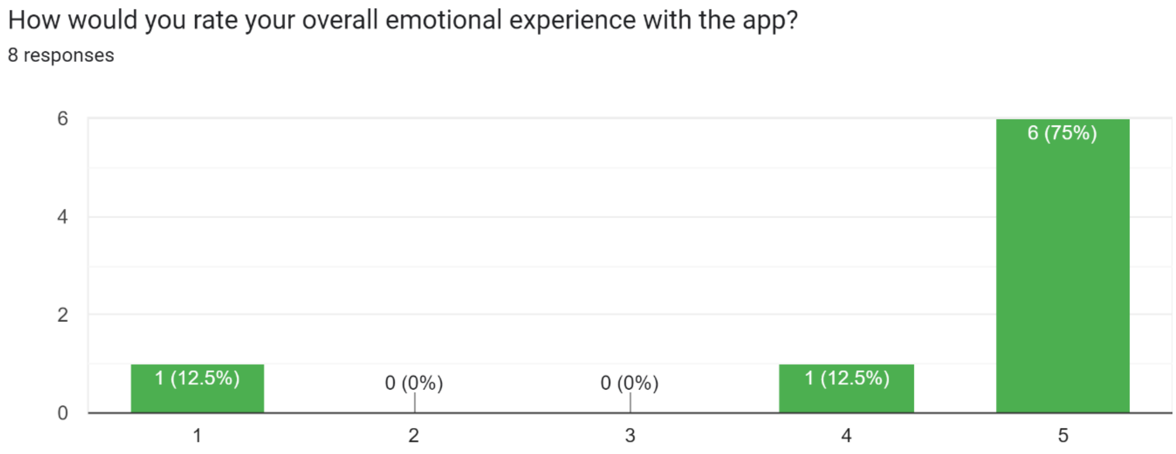

Figure 19: Participants Overall Emotional Experience

One participant indicated a score of very negative (1/5), another participant indicated a positive emotional experience (4/5) while 75% (6 participants) indicated the maximum of very positive (5/5). The very-negative scoring participant negative results indicated they felt a moderate amount of frustration, and an extreme level of upset while using the app. Although did not provide any additional comments. The facilitator notes reveals that the participant found the camera views challenging and experienced a bug during one of the stages. This participant showcases the potential of negative emotions caused by bugs in the learning app. Although other participants did also experience bugs (3/8 participants), these participants did not experience such negative emotions as a result.

4.1.6. Tait's Model

The penultimate section of the questionnaire evaluates the learning environment and support provided by the learning app in accordance with Tait's model. The results indicate whether the virtual environment provides support and guidance and allows students to complete the tasks independently and allows them to explore solutions.

The first question asked students to again use the Likert scale of 1-5 where to lower bound indicated 'Not at all', and the upper bound indicated 'Extremely well'. The results are illustrated in Figure 20: Participants' Responses Regarding Learning App's Guidance and Support.

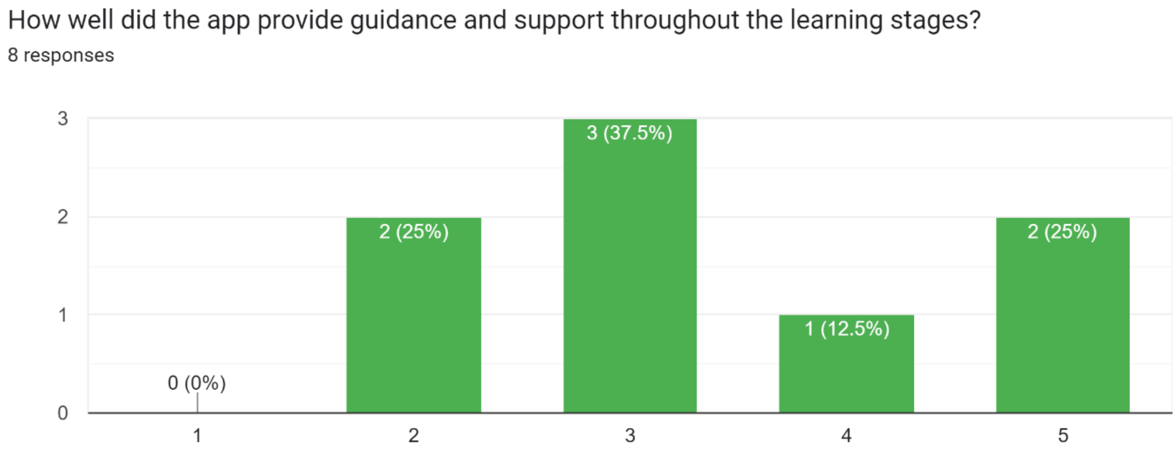

Figure 20: Participants' Responses Regarding Learning App's Guidance and Support

It is evident that a variety of responses was obtained. Additional comments suggested that the learning app itself did not provide any specific guidance and that it was "figure it out yourself", although participants did find it enjoyable as they were afforded the opportunity to create their own solutions from scratch. Participants also suggested that when starting with a level more interactive annotations or comments could be a valuable addition.

The following question featured the same Likert scale as before. Participants were asked whether the learning app allowed for independent exploration when completing the tasks. The results are illustrated in Figure 21: Participants' Responses Regarding Learning App's Independence Ability.

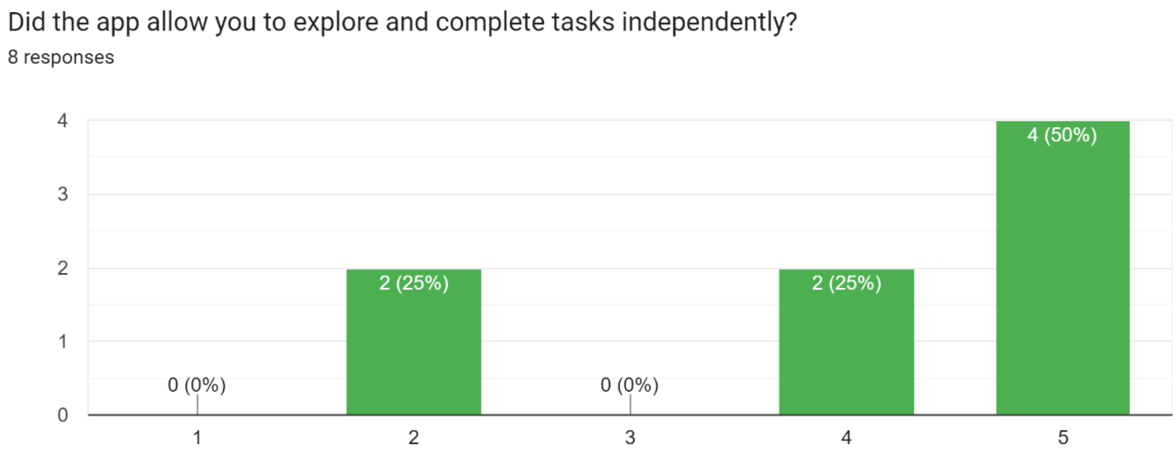

Figure 21: Participants' Responses Regarding Learning App's Independence Ability

The results show that 75% of participants indicated that the app provided independence in the high-extreme degree. While two participants indicated a low level of independence. Additional comments show that a facilitator that assisted when the participant was struggling or had questions was helpful.

4.1.7. Additional Feedback

The final section of the feedback questionnaire featured two questions that prompted participants to suggest improvements for the learning app to better facilitate learning, and to provide additional comments or suggestion regarding their experience with the learning app. All comments and suggestions are displayed in the appendix. The most common suggestion was to provide additional views or implement a free-look system as the currently available views were challenging during certain stages. Participants also indicated that additional instructions or "hints" withing the app that is customised for each stage could enhance the experience.

4.2. Observational Feedback

The facilitator recorded notes during the pilot study that detailed any disruptions, notable remarks and general observations of each participant. It was noted that several participants experienced disconnections in which the app would no longer communicate with the URSim controller, however no patterns were found that caused disconnections, besides that if the learning app received no input for an extended period of time it would occasionally require the facilitator to reconnect the two devices. Three participants experienced a bug where if a certain robot path was taken during level 4, the robot would pick up both balls and would not drop them. This bug was encountered during development and was believed to be addressed although, evidently, was not completely fixed. The facilitator also noted that almost all participants experienced challenges regarding the available camera views and that additional views were required. Additional notes also indicated that almost half of participants used the co-ordinate system controls of the native URSim controller, even though it was not included in the introduction.

4.3. Summary of Key Findings

An overview of key findings during the pilot study is detailed in this section to conclude the results section supported by Table 1: Summarized Results which contain the averages of Likert-scale responses regarding each framework's aspects. The learning app was successful in providing an effective learning experience that satisfied learning objectives.

	Aspect 1	Score 1	Aspect 2	Score 2	Aspect 3	Score 3
RBT	Learning Gained	4.69				
ADDIE	Usability and Instructional Design	4.31				
NASA TLX	Task Load	2.46	Perceived Performance	4.38		
PANAS	Positive	4.3	Negative	1.46	Overall	4.38
Tait	Guidance and Support	3.38	Independent Learning	4.00	Motivation	4.63
Interest in ICR	Before	2.88	After	4.38	Gained	1.5

Table 1: Summarized Results

Participants indicated an increase interest in ICRs after completing the pilot study. The RBT tiered model was satisfied on all tiers although further potential exists for the final tier, creating. The ADDIE model's final steps for development provided valuable feedback regarding the learning app and indicated that the proof-of-concept learning app was sufficient in design. Additional comments and suggestions were gathered that could be incorporated in future versions that would improve and refine the learning experience.

The perceived workload and task difficulty was evaluated using the NASA TLX model and indicated that although the tasks were challenging, they were not overly demanding or difficult. The effect of bugs was noted to be a huge factor in negative emotions such as frustration. The modified PANAS questionnaire indicated similar results regarding bugs, therefore extreme care should be exercised in future versions to ensure minimal bugs plague the app. Although the overall positive emotions far outweighed the negative emotions experienced by participants, this is an essential result for an effective learning tool, as effective learning can be hindered by negative emotions caused by the tool. The learning environment was evaluated using Tait's model. It was found that although the learning app provided a satisfactory environment, additional support can be integrated to produce a more effective guidance and support system in the learning app itself.

Chapter 5 - Discussion and Recommended Improvements

The findings of the pilot study indicates that the learning app was successful in providing an effective educational experience, achieving the primary learning objectives. Established educational models, such as the RBT, ADDIE, and Tait were satisfied. This indicates that the app could provide learning progression to engineering students. However, the study revealed opportunities for further development, especially in achieving the final tier of the RBT model and incorporating feedback gathered by the ADDDIE framework.

Despite the overall success of, many areas of improvement were revealed from the feedback, most notably the user experience. While the tasks exhibited the correct level of challenging bugs contributed to moments of frustration, as highlighted by both the NASA TLX workload assessment and the PANAS emotional scale. Addressing these bugs is crucial to maintain positive emotions and minimize negative emotions that is required for an effective educational tool. Tait's model affirmed the app provided a supportive learning environment, the study suggests that additional guidance and support features should be incorporated to improve the user experience.

5.1. Learning Objectives

In the following discussion, the app's effectiveness is evaluated, and recommendations are made for future iterations based on key findings from the pilot study. The learning objectives for which the learning app was designed to achieve included the introduction of ICRs and their role in industrial processes, the basics of ICR controls, writing simple pick-and-place programs, without the use of a physical robot while providing positive user engagement.

The feedback suggests that this was effectively satisfied in context of a demonstrative concept version however additional features are required for a classroom ready experience. Further iterations of the learning app and surrounding content should

- incorporate more advanced problems,
- that incorporate a larger variety of tasks
- with defined objectives and performance evaluation,
- featuring additional feedback mechanisms,
- supported by additional training and instructional material,
- while providing additional context to problems in the form of a case study or similar format to establish a sense of purpose.

5.2. User Experience and Usability

Overall user experience and usability of the learning app was satisfactory although improvement areas exist.

- Improved navigation can be achieved through a 'proceed to next level' button that appears upon completion of a level's requirements.
- A 'restart' button to reset the score and environment.
- Additional support using hints or annotations or intermediate tutorials.
- Improved collision feedback such as highlighting where the collision occurred in example.
- The addition of more camera angles/views or the addition of a 'free look' camera control.
- The refinement of the TCP protocol to avoid disconnections and provide a smoother experience.
- The elimination of bugs through thorough and rigorous bug testing.
- Refinement of object-pickup-system or rework to include options for actual tools to more accurately portray real world operations and robot utilization.

5.3. Cognitive and Emotional Impact

Participants reported a moderate level of mental demand while using the app, which suggests that the tasks were intellectually stimulating but manageable. The nature of the tasks required focused attention, but the complexity did not cause significant cognitive overload. This balance is important for maintaining engagement and ensuring that learners are pushed to develop their skills without becoming frustrated by an excessive workload. By scaffolding tasks, students could build on prior knowledge and skills, enabling learning without overwhelming them.

The PANAS feedback revealed that positive emotions, such as interest and excitement, were present in the learning process. However, some negative emotions were closely tied to technical performance of the app. Users expressed satisfaction with the overall learning experience, but negative emotional responses could potentially hinder the effectiveness of the app as a learning tool. Frustration was associated with the occurrence of bugs and unanticipated disconnections, which hindered the smooth flow of the learning process. This emotional response is crucial to address, as frustration can reduce user motivation and negatively impact learning outcomes.

Ensuring that the app operates without technical errors, especially those related to connection stability, is vital in minimizing negative emotional experiences.

5.4. Challenges and Limitations

A notable limitation of the pilot study is the small sample size, with only eight students participating in the testing of the learning app. Although their feedback provided valuable insights, a larger and more diverse pool of participants would provide more representative evaluation regarding the app's effectiveness and potential flaws. A more refined version of the learning app and its accompanying educational content should be tested on a larger scale, such as an entire class. This will likely result in a more accurate reflection of the app's usability, educational value, and scalability in a real academic environment.

The biggest challenge encountered during the development of the learning app was the instability of the TCP protocol, which caused frequent disconnections and disrupted the user experience. The communication protocol, inherited from Mr Parak's project that forms the foundation of this project, was challenging to adjust and optimize due to the complexity of network communication and the intricacies of maintaining stable connections between the learning app on the android tablet and the URSim software located on virtual machine on a separate device. Future iterations of the learning app require a rework or replacement of this protocol to achieve seamless functionality.

Chapter 6 - Conclusion

This project addressed the educational gap in the undergraduate engineering curriculum. The gap in question is defined as the lack of practical education in ICRs and the control and programming of these valuable Industry 4.0 tools. This project developed a proof-of-concept solution to supplement this gap accompanied by its associated learning objectives and challenges using a scalable serious game-like application that utilizes a native controller for ICRs by providing a hands-on and practical educational tool within a virtual environment. The app was developed using proven educational content creation methodologies and was subject to a pilot study to determine the effectiveness of the solution, the feedback questionnaire featured in the pilot study was developed using the educational content creation models.

The learning app and its accompanying learning material was highly effective as a proof-of-concept demonstration version of an educational tool that bridges the gap between theory and practical skills while engaging students in an interactive learning process. Although various improvement areas and additional features are suggested based on the feedback from the pilot study and development process alongside a larger and more complete pilot study before the tool can be implemented in the academic curriculum of undergraduate engineers.

This project however showcased the potential for virtual environments that utilize game-like structures as cost-effective and scalable alternatives to physical ICRs in the classroom. It is also noted that this solution could be adjusted and refined for other educational gaps with similar limitations and challenges of scalability and cost-feasibility, while creating an effective alternative that provides an engaging and effective learning environment.

Chapter 7 - Personal Reflection

The following section includes a personal reflection of the author and therefore will include first person language. This reflection discusses my experiences, challenges and lessons learned throughout my final year project.

My initial goals for the project included learning new software and methods that is used in creating game-like applications as computer games has always been a hobby of mine. I appreciate games as they have the ability to develop and enhance skills and often accredit my foundational level of problem solving and logistical mindset to games as they often comprise of challenges in order to advance to higher stages or to compete with or against friends. The second objective I had for this project was to create a proof-of-concept model that could truly be used in the engineering curriculum so that students could learn effectively in an engaging environment that was enjoyable and inspired interest for Industrial Engineering. South Africa is an amazing country with exceptional people and talent, therefore the lack of education due to constraints such a funding is a pressing issue. I find an extreme level of gratitude to be able to contribute to the solution of this problem as discussed in this project, al be it a minute part. I believe strongly that game-like applications such as the model developed possess great potential in improving and scaling educational material not only for the applied application of ICR's, but also for various other topics.

Initial research went smoothly although feedback obtained concerning the literature report was eye-opening. I realized that my technical communication and reporting was not up to standard, which was a key skill I believe I improved during this project. Development in Unity was not as I expected and therefore progress in the start was painfully slow and comprised of various instances where I believed that I would not be able to produce a working model. Although determination and hours of research and practice led to a new understanding of the software which resulted in faster pace and enjoyment while developing. A significant challenge that eluded me due to the complexity of the challenge and time constraints associated with this project was the network communication. Various issues before and during the pilot study originated from here and caused many issues. I regret not being able to fully optimise and refine this component, although am delighted to achieve a working state. This project has solidified my confidence in overcoming technical and emotional challenges, and I look forward to applying these skills in future engineering roles.

Chapter 8 - Bibliography

Al Madi, N., Peng, S. and Rogers, T. (2022) ‘Assessing Workload Perception in Introductory Computer Science Projects using NASA-TLX’, in Proceedings of the 53rd ACM Technical Symposium on Computer Science Education - Volume 1. New York, NY, USA: Association for Computing Machinery (SIGCSE 2022), pp. 668–674. Available at: https://doi.org/10.1145/3478431.3499406.

Amish, M. and Jihan, S. (2023) ‘Developing a Virtual Engineering Lab Using ADDIE Model’, Journal of e-learning Research, 2(1), pp. 50–69. Available at: https://doi.org/10.33422/jelr.v2i1.417.

Bloss, R. (2016) ‘Collaborative robots are rapidly providing major improvements in productivity, safety, programming ease, portability and cost while addressing many new applications’, Industrial Robot: An International Journal, 43(5), pp. 463–468. Available at: https://doi.org/10.1108/IR-05-2016-0148.

Cao, A. et al. (2009) ‘NASA TLX: Software for assessing subjective mental workload’, Behaviour Research Methods, 41(1), pp. 113–117. Available at: https://doi.org/10.3758/BRM.41.1.113.

Chigbu, B.I. and Nekhwevha, F.H. (2022) ‘The collaborative work experience of robotics and human workers in the automobile industry in South Africa’, African Journal of Science, Technology, Innovation and Development, 14(1), pp. 280–287. Available at: https://doi.org/10.1080/20421338.2020.1837446.

Colgate, J.E., Wannasuphoprasit, W. and Peshkin, M.A. (1996) ‘Cobots: Robots for collaboration with human operators’, Dynamic Systems and Control. Available at: https://doi.org/10.1115/imece1996-0367.

Collaborative Robots and Industrial Revolution 4.0 (IR 4.0) (no date). Available at: https://ieeexplore.ieee.org/abstract/document/9080724 (Accessed: 4 September 2024).

CoppeliaSIM User manual (no date) Coppeliasim User Manual. Available at: https://www.coppeliarobotics.com/helpFiles/ (Accessed: 10 May 2024).

Craig, J.J. (2005) ‘Introduction to robotics: Mechanics and Control’, in Introduction to robotics: Mechanics and Control. Third Edition, pp. 4–16.

Dewa11, M.T., Adams, D.Q., Nyanga, L., Gxamza, M. and Ganduri, L., 2018. Industry 4.0: A myth or a reality in South Africa?

Faccio, M., Bottin, M. and Rosati, G. (2019) ‘Collaborative and traditional robotic assembly: a comparison model’, The International Journal of Advanced Manufacturing Technology, 102(5), pp. 1355–1372. Available at: https://doi.org/10.1007/s00170-018-03247-z.

Find your universal robots distributor (no date) Collaborative robotic automation. Available at: https://www.universal-robots.com/distributors/ (Accessed: 10 May 2024).

Forehand, M., 2010. Bloom’s taxonomy. *Emerging perspectives on learning, teaching, and technology*, *41*(4), pp.47-56. Available at: https://www.d41.org/cms/lib/IL01904672/Centricity/Domain/422/BloomsTaxonomy.pdf

Ganesan, M. (2015) ‘Developing of E-content package by using ADDIE Model’, Journal of Learning and Development, pp. 52–54.

George, D.A.S. and George, A.S.H. (2023) ‘The Cobot Chronicles: Evaluating the Emergence, Evolution, and Impact of Collaborative Robots in Next-Generation Manufacturing’, Partners Universal International Research Journal, 2(2), pp. 89–116. Available at: https://doi.org/10.5281/zenodo.8021406.

Goel, R. and Gupta, P. (2019) ‘Robotics and Industry 4.0’, in. Available at: https://doi.org/10.1007/978-3-030-14544-6_9.

Groover, M.P. (2007) ‘Automation, Production Systems, and Computer-Integrated Manufacturing’, in Automation, Production Systems, and Computer-Integrated Manufacturing. 4th Edition, pp. 210–256.

Ibn Tofail University, National School of Applied Sciences, Engineering Sciences Laboratory, Kenitra, Morocco et al. (2024) ‘Maximizing efficiency and collaboration: Comparing Robots and Cobots in the Automotive Industry – A Multi-Criteria Evaluation Approach’, International Journal of Industrial Engineering and Management, 15(3), pp. 238–253. Available at: https://doi.org/10.24867/IJIEM-2024-3-360.

Javaid, M. et al. (2021) 'Substantial capabilities of robotics in enhancing industry 4.0 implementation', Cognitive Robotics, 1, pp. 58–75. Available at: https://doi.org/10.1016/j.cogr.2021.06.001.

Lasi, H. et al. (2014) 'Industry 4.0', Business & Information Systems Engineering, 6(4), pp. 239–242. Available at: https://doi.org/10.1007/s12599-014-0334-4.

Maisiri, W. and van Dyk, D.L. (2019) 'Industry 4.0 readiness assessment for South African industries', South African Journal of Industrial Engineering, 30(3), pp. 134–148. Available at: https://doi.org/10.7166/30-3-2231.

Mouhib, H. et al. (2024) 'Maximizing efficiency and collaboration: Comparing Robots and Cobots in the Automotive Industry – A Multi-Criteria Evaluation Approach', International Journal of Industrial Engineering and Management, 15(3), pp. 238–253. Available at: https://doi.org/10.24867/IJIEM-2024-3-360.

Mpungose, S.C. (no date) 'How SMEs in South Africa can implement robotics to increase competitiveness'. Available at: https://wiredspace.wits.ac.za/items/5252c1a1-8f4d-4b57-8125-46166c192806 (Accessed: 11 September 2024).

National development plan 2030 (no date) National Development Plan 2030 | South African Government. Available at: https://www.gov.za/issues/national-development-plan-2030 (Accessed: 10 May 2024).

Shin, H. et al. (2019) 'A real-time human-robot Collision Safety Evaluation Method for Collaborative Robot', 2019 Third IEEE International Conference on Robotic Computing (IRC). Available at: https://doi.org/10.1109/irc.2019.00106.

Tait, A. (2000) 'Planning student support for Open and distance learning', Open Learning: The Journal of Open, Distance and e-Learning, 15(3), pp. 287–299. Available at: https://doi.org/10.1080/713688410.

The future African workplace: the use of collaborative robots in manufacturing (no date). Available at: https://doi.org/10.4102/sajhrm.v15i0.901.

Unhelkar, V.V. et al. (2018) 'Human-aware Robotic assistant for Collaborative Assembly: Integrating Human Motion Prediction with planning in Time', IEEE Robotics and Automation Letters, 3(3), pp. 2394–2401. Available at: https://doi.org/10.1109/lra.2018.2812906.

Universal Robots Academy. Available at: https://academy.universal-robots.com/ (Accessed: 08 May 2024).

Universal Robot Cobot Cost Analysis and return on Investment: Electromate Inc (no date) Electromate Inc. Available at: https://www.electromate.com/resources/cobot-cost-analysis-and-roi/ (Accessed: 08 May 2024).

Watson, D., Clark, L.A. and Tellegen, A. (1988) 'Development and validation of brief measures of positive and negative affect: The panas scales.', Journal of Personality and Social Psychology, 54(6), pp. 1063–1070. Available at: https://doi.org/10.1037//0022-3514.54.6.1063.

Xu, X. et al. (2021) 'Industry 4.0 and Industry 5.0—Inception, conception and perception', Journal of Manufacturing Systems, 61, pp. 530–535. Available at: https://doi.org/10.1016/j.jmsy.2021.10.006.

Chapter 9 - Appendices

9.1. Appendix A: Abstract by Meyer van Dyk, Clint Alex Steed for 15th Conference on Learning Factories 2025.

15th Conference on Learning Factories, CLF 2025

Scaling Industrial Robotics Education: A Serious Game Approach for Resource-Limited Regions

Meyer van Dyk[a], Clint Alex Steed[a], *

Abstract

Industrial robots have been crucial in manufacturing but remain underutilized in South Africa due to historical wage factors and a lack of exposure. In education, large class sizes made practical exposure to robots costly and logistically challenging. This study developed a low-cost, scalable serious game for industrial robot programming, aimed at undergraduate engineering students. The game addressed the challenge of limited resources while providing students with hands-on experience using only a laptop or Android tablet. The virtual environment, built using the Unity engine, is connected to a universal robot controller via TCP, allowing students to interact with robots using their native programming interface. The game guided students through structured stages, simulating real-world industrial applications and progressively enhancing their programming skills. A pilot study evaluated the platform's usability, learning outcomes, and student confidence. It was found that students experienced realistic robot behavior without the need for expensive physical robots. The feedback from this study informed future improvements to the game, contributing to more effective teaching methods in robotics. This solution addressed the skills deficit in South Africa by introducing students to industrial robot programming and Industry 4.0 concepts. By simulating robot behavior, the platform increased safety, reduced the number of robots required, and optimized physical robot utilization. The virtual environment offered a scalable, cost-effective alternative to traditional robotics education, helping to bridge the technical skills gap in resource-limited regions.

Peer Review statement: Peer-review under responsibility of the scientific committee of the 15th Conference on Learning Factories 2025.

Keywords: Industrial robotics; Serious games; Engineering education; Industry 4.0 skills; Low-cost robotics training

9.2. Appendix B: Unity Model

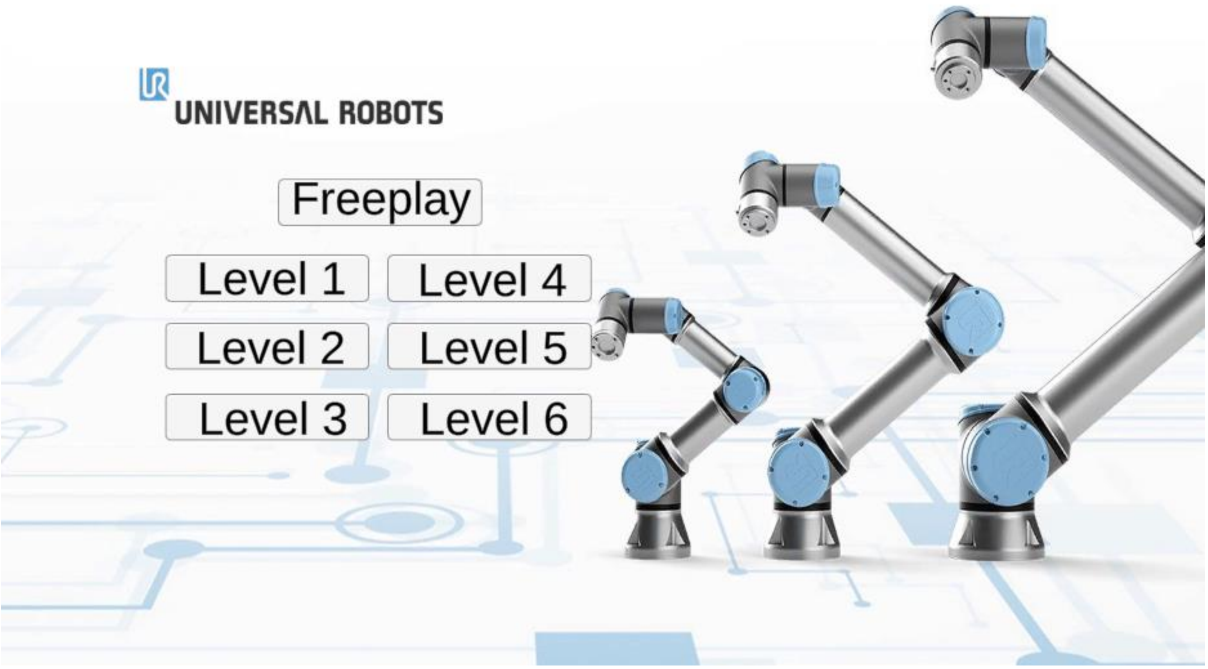

Figure 22: Main Menu User Interface

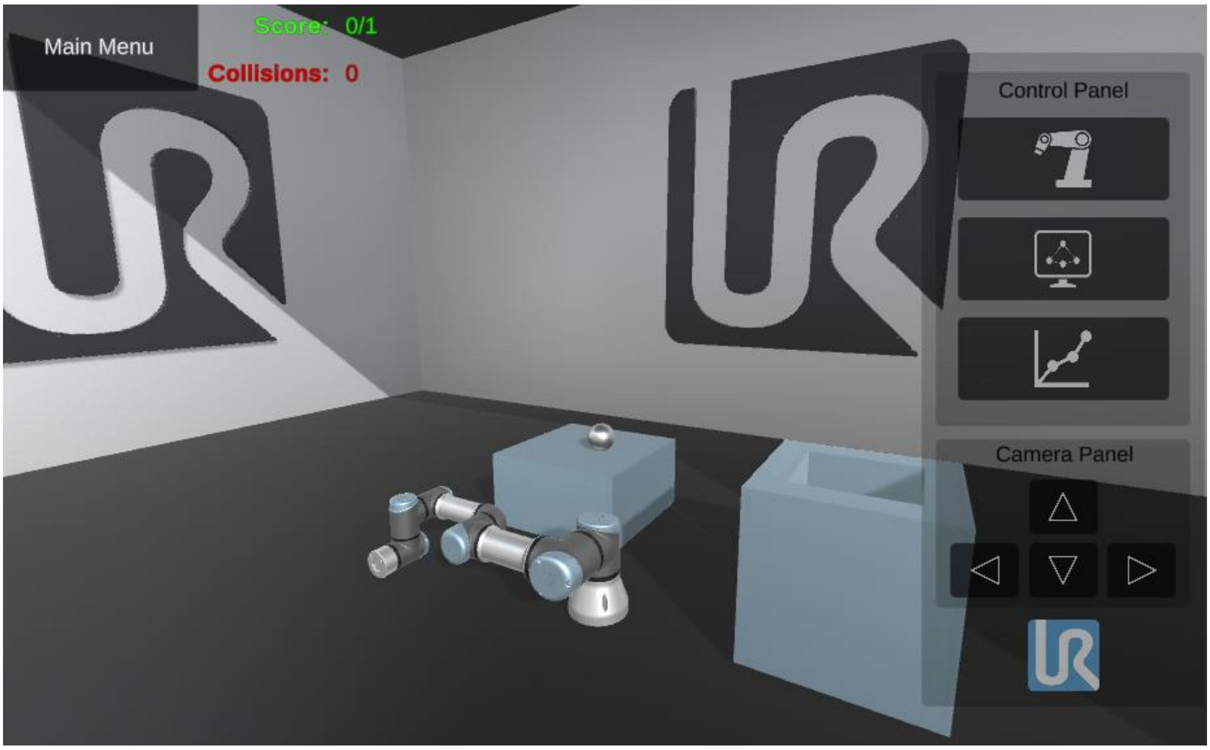

Figure 23: Level 1

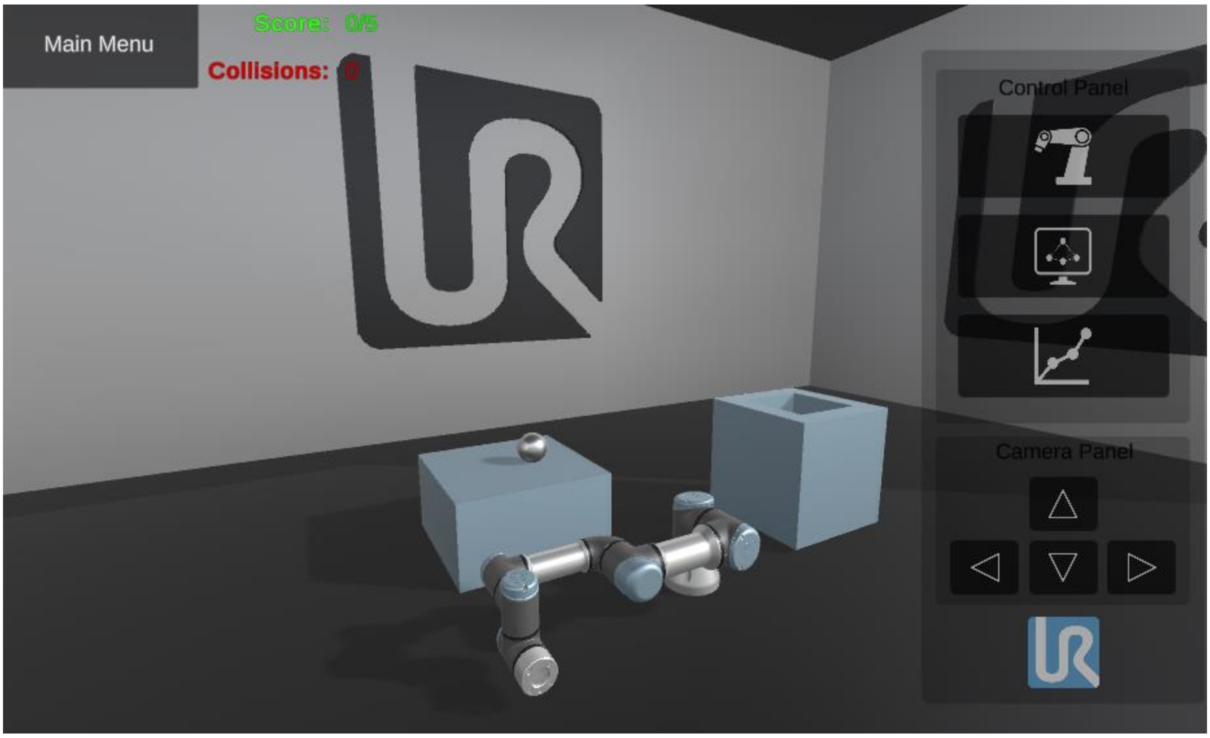

Figure 24: Level 2

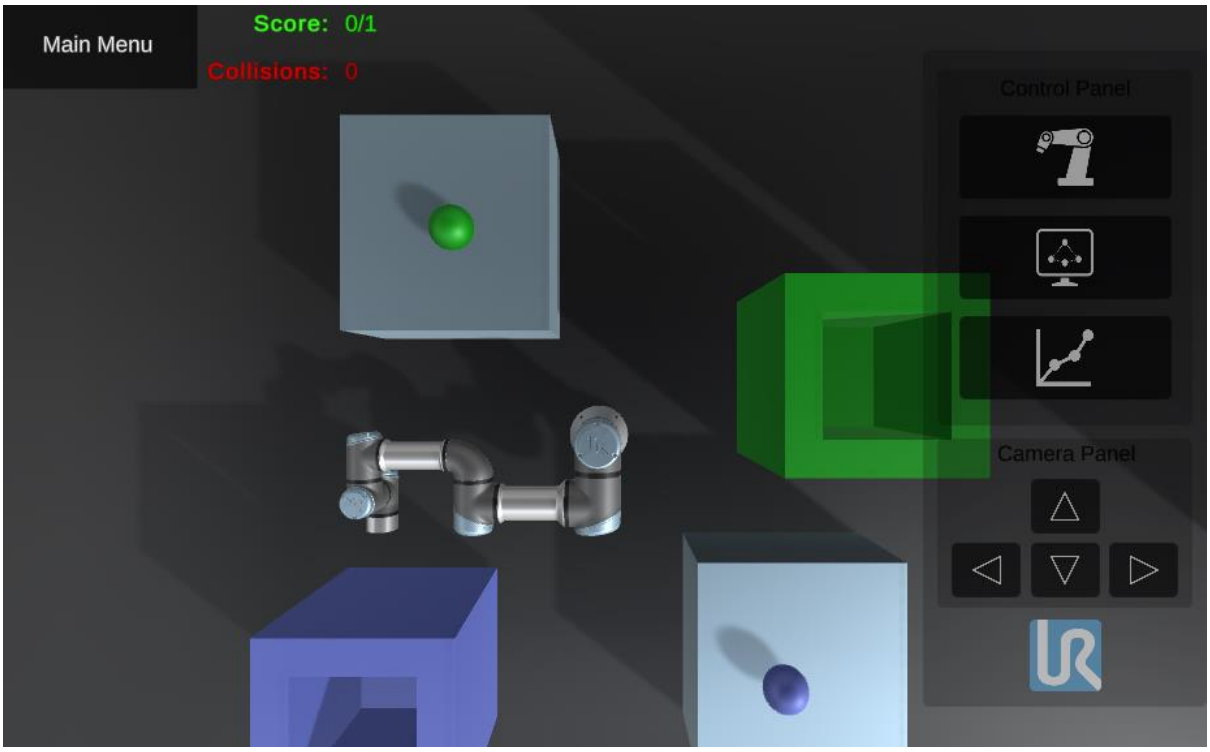

Figure 25: Level 3

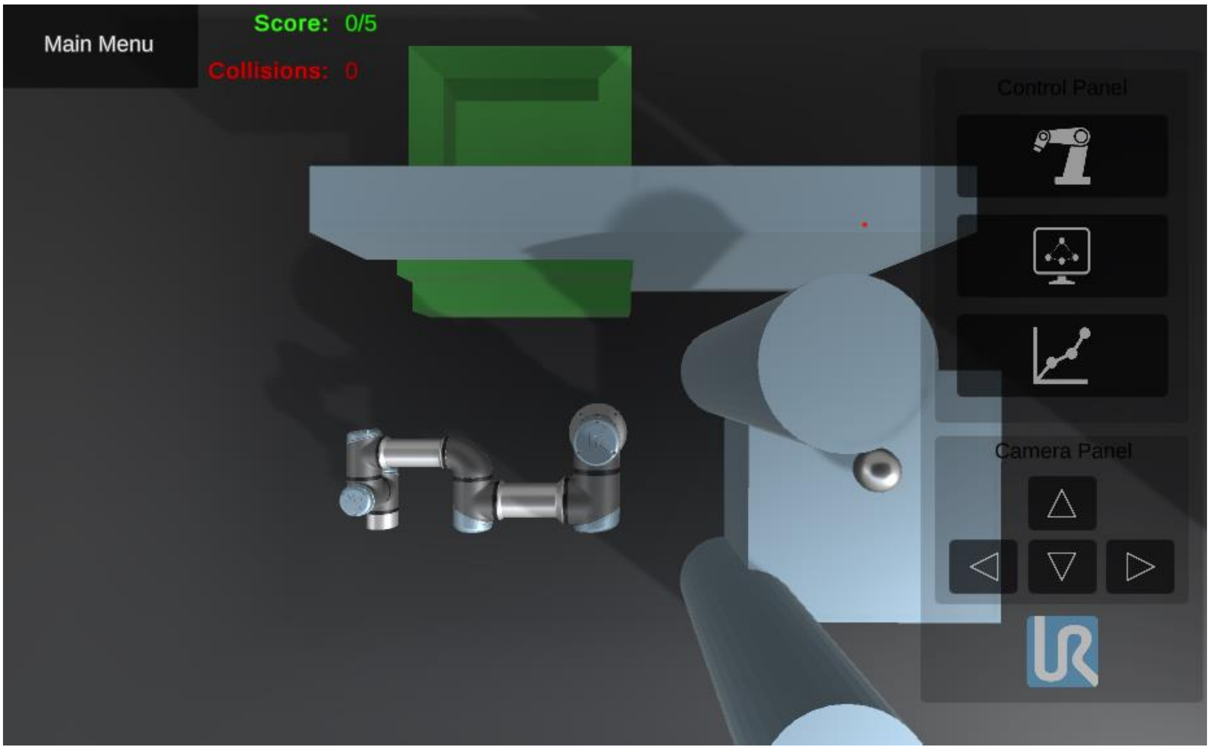

Figure 26: Level 6

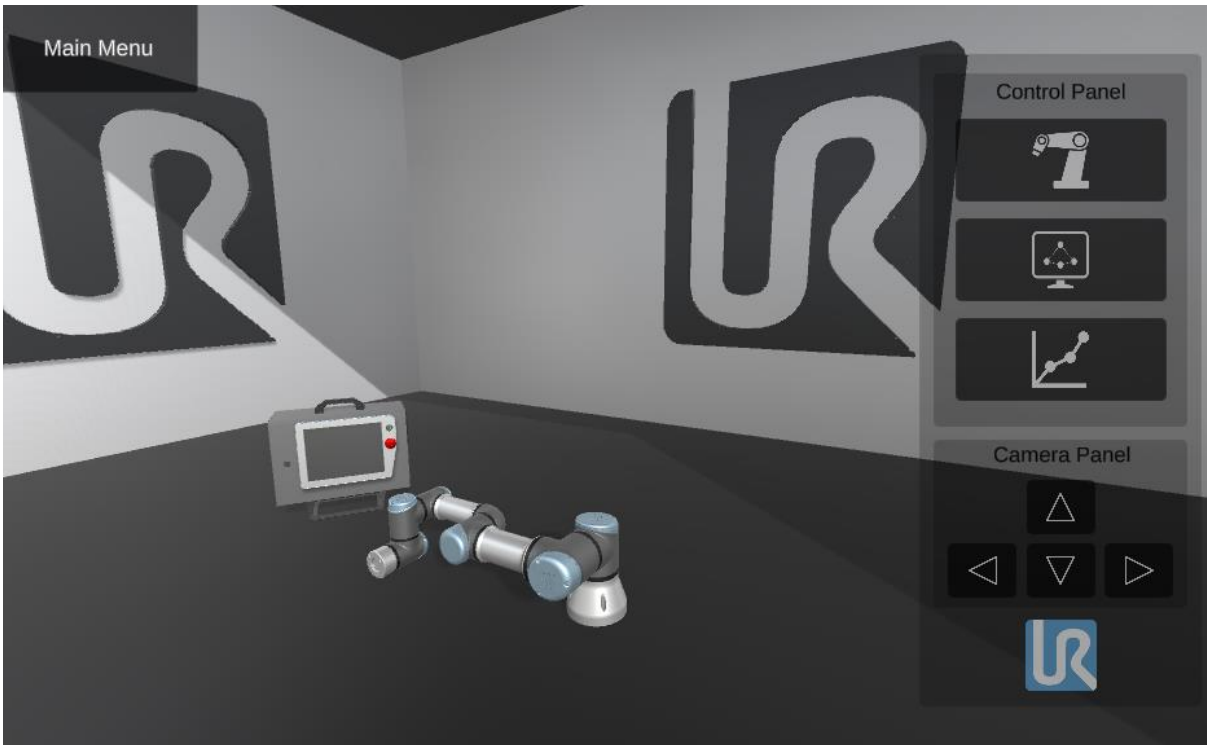

Figure 27: Freeplay Mode

9.3. Appendix C: Software Used

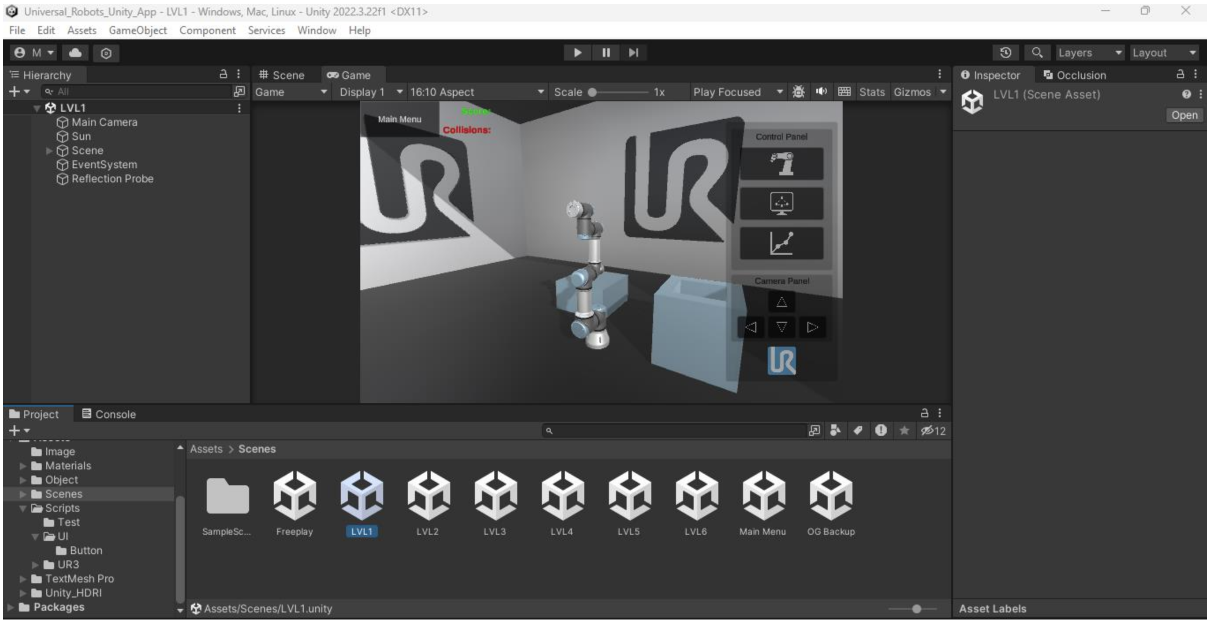

Figure 28: Unity Game Engine

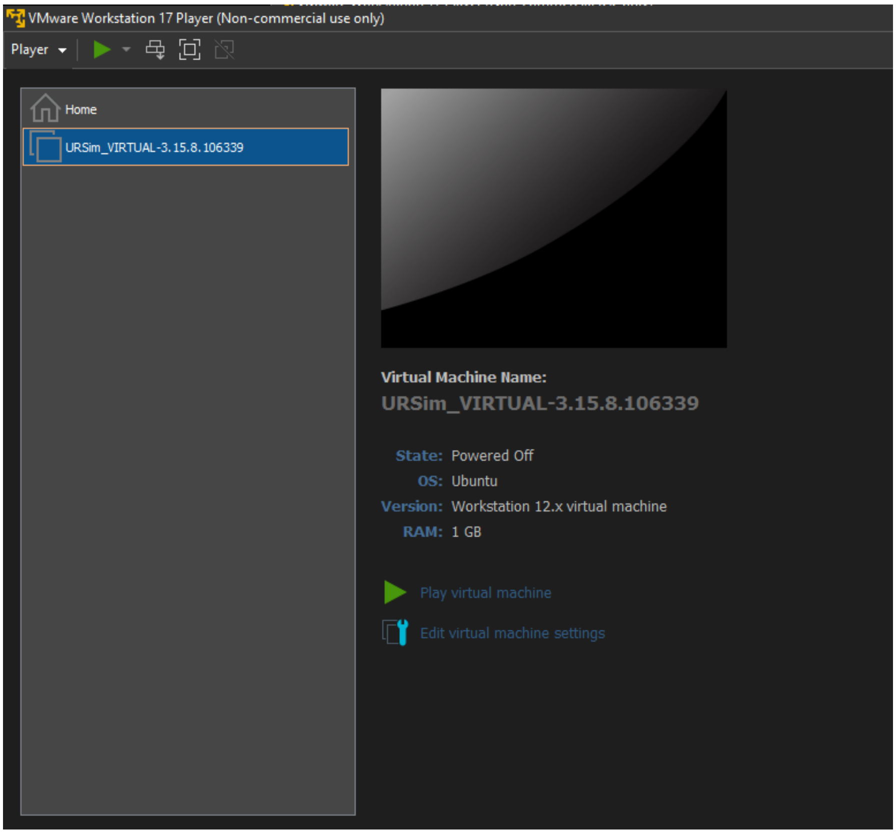

Figure 29: VMware Workstation 17 Player

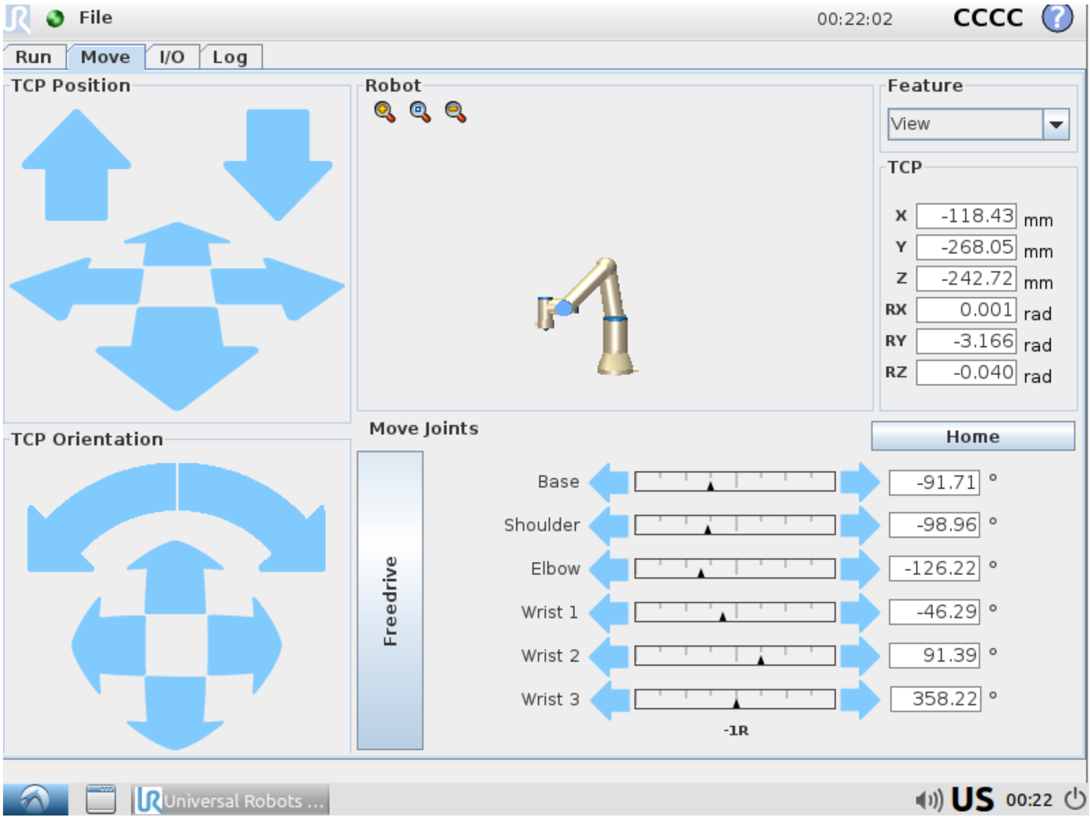

Figure 30: URSim Robot Controller

9.4. Appendix D: Pilot Study Questions and Results

Question	Averages	Participant 1	2	3	4	5	6	7	8
How effectively were you able to recall specific terms, functions, or commands related to the industrial robot during the tasks in the learning app?	4.375	4	5	4	5	4	5	4	4
How well did the learning app help you understand the core concepts of ICR programming?	4.625	5	5	4	5	4	5	4	5
Were you able to apply your knowledge to perform tasks involving the robot's operation and programming within the learning app?	5.000	5	5	5	5	5	5	5	5
Did the tasks in the learning app encourage you to break down the tasks and analyze the robot's movements to solve the task?	4.875	5	5	5	5	4	5	5	5
How well did the learning app allow you to evaluate the effectiveness of your task performance and programming solutions?	4.875	5	5	4	5	5	5	5	5
To what extent did the learning app enable you to develop new programming strategies during the tasks?	4.375	4	5	5	5	3	5	5	3
	4.688								

Table 2: Revised Bloom's Taxonomy Questionnaire Section Results

Question	Average	Participant 1	2	3	4	5	6	7	8
Was the layout of the app easy to navigate and understand?	3.25	1	5	1	5	3	1	5	5
Were the instructions clear and helpful for completing the tasks?	4.500	5	5	4	5	3	5	4	5
Did the app hold your attention and keep you engaged throughout the learning process?	4.625	5	5	5	5	4	5	5	3
How effective were the feedback tools (Score and Collision counters) in helping you monitor your learning?	4.875	5	5	5	5	5	5	4	5

Table 3: ADDIE Model Questionnaire Section Results

Questions	Averages	Participant 1	2	3	4	5	6	7	8
How mentally demanding did you find the tasks?	2.500	4	1	3	2	1	3	3	3
How much effort did you have to invest to complete the tasks?	3.000	4	2	3	4	2	3	3	3
Did you experience any frustration while completing the tasks?	1.875	2	1	2	1	5	1	1	2
	2.458								
How satisfied are you with your overall performance?	4.375	4	5	4	5	3	5	5	4

Table 4: NASA TLX Workload Assessment Questionnaire Section Results

Questions	Averages	Participant 1	2	3	4	5	6	7	8
Positive	**4.300**								
Level of Interest while using the app?	4.375	5	5	4	5	2	5	5	4
Level of Excitement while using the app?	4.000	5	5	4	4	2	4	4	4
Level of Inspiration while using the app?	4.375	5	4	4	5	3	5	5	4
Level of Determination while using the app?	4.250	5	5	5	4	4	4	4	3
Level of Attentiveness while using the app?	4.500	5	5	5	5	3	4	5	4
Negative	**1.458**								
Level of Nervousness experienced while using the app?	1.500	1	1	2	1	1	1	4	1
What level of upset did you feel while using the app?	1.750	1	1	1	1	5	1	3	1
What level of distress did you feel while using the app?	1.125	1	1	1	1	1	1	2	1
What level of frustration did you feel while using the app?	1.625	2	1	1	1	3	1	3	1
What level of fear did you feel while using the app?	1.250	1	1	1	1	1	2	2	1
What level of confusion did you feel while using the app?	1.500	1	1	2	1	1	2	2	2
Overall									
How would you rate your overall emotional experience with the app?	**4.375**	5	5	4	5	1	5	5	5

Table 5: Modified PANAS Questionnaire Section Results

Questions	Averages	Participant 1	2	3	4	5	6	7	8
How well did the app provide guidance and support throughout the learning stages?	3.375	3	4	2	5	3	5	2	3
Did the app allow you to explore and complete tasks independently?	4.000	5	5	4	2	2	4	5	5
How motivated were you to complete the tasks in the app?	4.625	5	5	5	5	4	5	5	3
	4.000								

Table 6: Tait's Model Questionnaire Section Results

Additional									
Interest before	2.875	2	5	3	3	2	2	5	1
Interest after	4.375	5	5	4	5	3	5	5	3
Increase (after - before)	**1.500**	3	0	1	2	1	3	0	2

Table 7: Additional Feedback Questionnaire Section Results

9.5. Appendix E: Unity Coding (C#) of Critical Interactions

```
0 references
void Start()
{
    isPickedUp = false;
    magnetPoint.GetComponent<SphereCollider>().enabled = false;
}
    0 references
    private void OnTriggerEnter(Collider other){


    // pickup if it has tag "PICKUPABLE" and we are not carrying anything.
    if (isPickedUp == false){

    if (other.transform.CompareTag("PICKUPABLE"))
    {
        currentItem = other; // Set var currentItem to object picked up.

        Debug.Log("PickedUp");

        Rigidbody rb = other.GetComponent<Rigidbody>(); // Gets and assigns object's rigidbody component to var rb
        rb.isKinematic = true; // Turns off physics of object.

        other.transform.position = magnetPoint.transform.position; // Attach object to magnet point.

        other.transform.SetParent(magnetPoint.transform); // Let object follow magnet point.
        magnetPoint.GetComponent<SphereCollider>().enabled = true;
```

Figure 31: Pick-Up Code

```
1 reference
public void releaseball(){
            try
            {
            isPickedUp = true; //release object.
            magnetPoint.GetComponent<SphereCollider>().enabled = false;
            Rigidbody rb = currentItem.GetComponent<Rigidbody>(); // Gets and assigns object's rigidbody component.
            rb.isKinematic = false; //Turns on objects physics.
            currentItem.transform.SetParent(null); //Disables object following the magnet point.
            }
            catch (System.Exception)
            {
                Debug.Log("No ball to drop");
                isPickedUp = false;
                magnetPoint.GetComponent<SphereCollider>().enabled = false;
            }
 }
```

Figure 32: Release Code

```
public void IncreaseScore()
{
    BallsInCorrectZone += 1 ; // Increase ball count in correct zones

    if(BallsInCorrectZone == NumberOfBalls) //Check if all  balls in correct zones, increase score.
    {
    ScoreNumber = ScoreNumber + 1; // Increase Score
    UpdateScoreDisplay();
    BallsInCorrectZone = 0; // Reset amount of objects in goalzones after respawning.

    // Find all objects with the "PICKUPABLE" tag
    GameObject[] balls = GameObject.FindGameObjectsWithTag("PICKUPABLE");

    // Loop through each ball and trigger respawn
    foreach (GameObject ball in balls)
    {
        ball.GetComponent<BallRespawn>().RespawnBall();
    }
    }
}
```

Figure 33: Score Counter Code

```
1 reference
public void RespawnBall(){
    Debug.Log("Respawn");

                    Rigidbody rb = GetComponent<Rigidbody>();
                    rb.velocity = Vector3.zero; //set velocity to 0
                    rb.angularVelocity = Vector3.zero; //set angular velocity to 0 to eliminate ball spin when spawning

                   transform.position = RespawnPoint; //respawns ball
}
0 references
private void OnTriggerEnter(Collider other){

    if (other.transform.CompareTag("RespawnTriggerZone"))
            {


                magnet.GetComponent<PickupBlock>().isPickedUp = false; //Allow magnet to pick up a new ball.

                if(TargetRespawnZone.GetComponent<BoxCollider>() == other) //Check if ball is in correct end zone, then update score.
                {
                ScoreCard.GetComponent<ScoreCard>().IncreaseScore();
                }
                else
                {
                RespawnBall(); //Respawn Ball if wrong box
                }

            }
}
```

Figure 34: Respawn Ball Code

9.6. Appendix F: Example of URSim Robot Programming

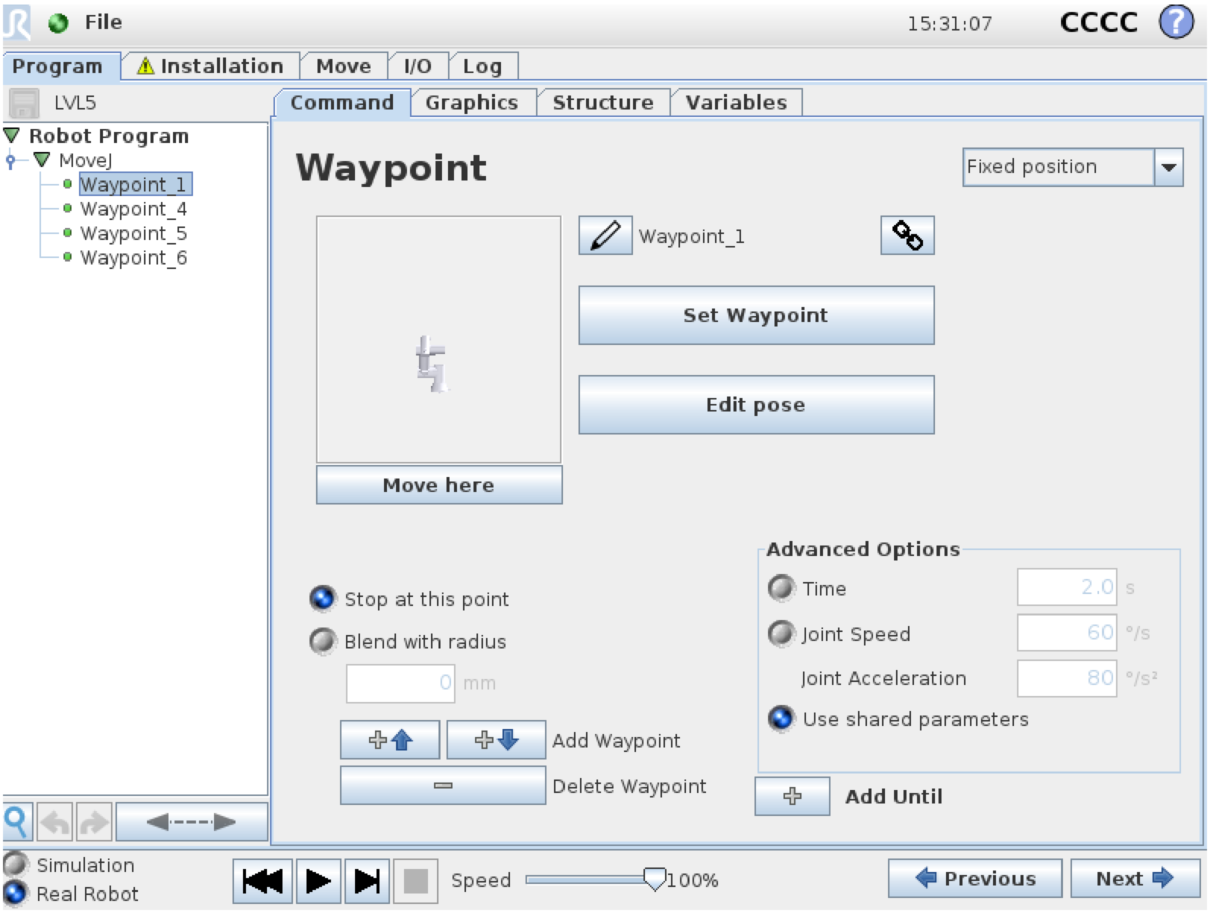

Figure 35: Example of Level 5's Robot Program Solution

9.7. Appendix G: Project Plan

09/04	Project Proposal Submission.	Completed
22/04	Feedback on the proposal. The first version of Digital Twin.	Completed
17/05	Literature report.	Completed
28/06	Finalization of the base app and twin. Start planning for challenges/problems.	Completed
05/07	Finalization of challenges/problems planning.	Completed
16/08	Challenges/problems to be done and testing to start.	Completed
16/09	70% Draft submission. Finalization of all technical work.	Completed
07/10	Final draft submission.	Completed
28/10	Final report submission.	Completed
18/11	Oral examinations start.	Completed

Table 8: Project Plan