AUGMENTED REALITY IN EDUCATION FOR ASSISTING STUDENTS OF INTERFACE COMPUTER COLLEGE ILOILO

A Thesis presented to the Faculty of the College of Computer Studies Interface Computer College, Inc.

Iloilo Campus

In Partial Fulfillment of the Requirement for the

Degree of Bachelor of Science in Computer Engineering and Bachelor of Science in Computer Science

By:

Clarence Lovely D. Panizales

Jhon Prietse D. Tacaisan

Lindsay Ruivivar

June 2023

APPROVAL SHEET

The Thesis entitled **AUGMENTED REALITY IN EDUCATION FOR ASSISTING STUDENTS OF INTERFACE COMPUTER COLLEGE ILOILO** prepared and submitted by **Clarence Lovely D. Panizales, Jhon Prietse D. Tacaisan, and Lindsay Ruivivar** in partial fulfillment of the requirements of the degree of Bachelor of Science in Computer Engineering and Bachelor of Science in Computer Science, has been examined and recommended for acceptance and approval for the oral examination.

Benedic C. Mallari, MIT

Adviser

PANEL OF EXAMINERS

Benedic C. Mallari, MIT

Panel Member

Engr. Henry C. Campullo

Chairman

Accepted in partial fulfillment of the requirements for the degree of Bachelor of Science in Computer Engineering and Bachelor of Science in Computer Science.

June 2023
Date

Ms. Sylvia A. Villarosa
Asst. School Administrator/Principal

ACKNOWLEDGEMENT

The developer would like to express their heartfelt gratitude and appreciation to the individuals who played a significant role in making this study possible.

First and foremost, the developer would like to express their sincere gratitude to God Almighty for providing grace and blessings throughout every step of the research process.

Special acknowledgment is given to Interface Computer College Inc. Iloilo staff for their cooperation, hospitality, and eagerness to assist during the study.

The developers thank those who served as critics by offering valuable suggestions, corrections, and comments. Their insights and explanations were instrumental in this study.

The developers are also grateful to their subject teachers who showed understanding, support, and advice throughout the research journey.

Thank you to the developer's parents for their financial, moral, emotional, and spiritual support. This has been invaluable in finishing this study.

The developers would like to express their deep appreciation to their thesis adviser, Mr. Benedic Mallari, for managing the panel, assisting with the manuscript, and dedicating time from his busy schedule to guide and cater to the developer's needs.

Lastly, the developers are grateful to the panelists, Engr. Henry C. Campullo and Mr. Benedic C. Mallari, MIT, for evaluating the program and providing valuable opinions and suggestions to enhance this thesis proposal.

The developers recognize and appreciate the invaluable contributions and support of all the individuals mentioned above. Their assistance, encouragement, and inspiration were crucial to the completion of this study.

DEDICATION

In this thesis, the developer dedicates their academic journey to:

To Almighty God, for guiding them every step of the way. The developers express their deepest gratitude for the blessings and strength that have sustained them.

The developer's beloved parents, for their unwavering love, guidance, and sacrifices. Their support and belief in the developer's abilities are their driving force.

Interface Computer College Inc. faculty, whose dedication and passion for knowledge have shaped the developer's intellectual growth. The developers are grateful for their knowledge and opportunities.

The developer's friends and colleagues, for their unwavering support, camaraderie, and shared experiences. Their encouragement and friendship have made the developer's journey meaningful and enjoyable.

The developers sincerely thank and appreciate all who have played a significant role in their academic success. Their support and belief in the developers are integral to their journey.

ABSTRACT

THESIS TITLE	**:**	Augmented Reality in Education for Assisting Students of Interface Computer College Iloilo
RESEARCHERS	**:**	Panizales, Clarence Lovely D. Ruivivar, Lindsay Tacaisan, Jhon Prietse D.
TYPE OF DOCUMENT	**:**	Thesis
TOTAL NO. OF PAGES	**:**	41
DEGREE	**:**	Bachelor of Science in Computer Engineering and Bachelor of Science in Computer Science
ADVISER	**:**	Benedic C. Mallari, MIT
SCHOOL NAME	**:**	Interface Computer College Inc. - Iloilo Campus
SCHOOL ADDRESS	**:**	3/F Emmanuel Business Center, Benigno Aquino Avenue, Diversion Road, Molo, Iloilo City
YEAR COMPLETED	**:**	2023
SUMMARY	**:**	

This abstract explores the use of augmented reality (AR) technology at Interface Computer College Inc. Iloilo - Campus to enhance student learning. AR offers immersive experiences that complement traditional teaching methods. It highlights AR's benefits in improving learning outcomes, student engagement, and collaboration. The abstract acknowledges challenges and discusses future potential.

AR overlays digital content in the real world, creating an engaging learning experience. It allows students to visualize complex concepts, interact with 3D models, and manipulate virtual objects. This interactive approach deepens understanding and promotes active learning.

Interface Computer College Inc. Iloilo - Campus provides real-time guidance, feedback, and support tailored to their needs. It offers step-by-step instructions, highlights key information, and explains concepts. AR also enables collaborative learning, allowing students to work together remotely on projects and problem-solving activities.

Implementing AR may face challenges such as technical infrastructure, costs, and access to devices. Integration and alignment with the curriculum require careful planning and training.

AR holds promise for Interface Computer College Inc. Iloilo - Campus. Advancements will lead to more sophisticated applications, personalized learning, and collaboration. Collaboration among educators, experts, and researchers will refine AR's potential.

In conclusion, AR technology enhances student learning at Interface Computer College Inc. Iloilo - Campus. It offers an immersive and interactive experience that improves outcomes and fosters collaboration. Continued exploration and integration of AR can lead to innovative teaching methods and prepare students for the Digital Age.

Table of Contents

List of Tables

List of Figures

CHAPTER I

INTRODUCTION

As part of the paper, this chapter introduces the problem and its context. Among the information included is the background of the study, the statement of the problem, the study's objectives, the conceptual framework, the scope of the study, limitations, and significance.

Background of the Study

The use of technology in the teaching and learning process has become increasingly popular in modern educational institutions as an attempt to enrich educational practices that occur in a school environment. AR technology allows the combination of real objects and virtual information in order to increase students' interaction with physical environments and facilitate their learning. Students can use real-world imagination to learn and understand difficult concepts, unlike other computer interfaces that draw users away from the real world and onto the screen. In addition, it will allow them to understand them better as they are experiencing concepts in the practical, real-time world. With the hands-on learning approach provided by AR, students can achieve better knowledge retention and deepen their understanding of a specific subject. Developing technology enables students to learn complex topics fun and easily through virtual reality devices. In addition, it will enhance the educational experience by bringing it closer to the senses and making it more tangible than just abstract thinking and creates opportunities for teachers to help students grasp abstract concepts.

AR motivates students to engage, and interact and increases the ability of students to learn curricular subjects. This will enable students to learn faster and memorize information. For this, Devices, equipment, and materials are used by the teacher to enhance the teaching-learning process in order to facilitate learning on the part of the students, simplifying concepts for young learning, fulfill their immediate needs, and motivate them towards more engagement in the learning process, in addition to reinforcing the relationship between them and their teacher. (Salama, 2005, Khalaf, 1985,)

Augmented reality (AR) is a new technology that has emerged with the potential for application in education. While a lot of research has been conducted on AR, few studies have been conducted in the education field. AR can reinvigorate the classroom, generating student excitement and encouraging course content engagement. More engagement allows students to take a more active role in their learning. Augmented reality asks students to become active participants in their own learning, and it can make them more interested and engaged in the subject matter. AR learning provides a gamified learning approach, making the lessons fun. As a result, it serves a positive impact on the students. The number of studies on AR is growing due to the effectiveness of this technology in recent years. AR has been used in different fields of education. In order to better prepare students for the future, augmented reality (AR) in education includes elements that improve the formation of abilities like problem-solving, teamwork, and invention. This medium provides distinctive affordances by combining physical and virtual worlds and continuous and implicit user control of the point of view and interactivity. It is also beneficial for conventional schooling that emphasizes technical skills and knowledge.

In particular, AR provides an efficient way to represent a model that needs visualization (Singhal et al., 2012). AR also supports the seamless interaction between the real and virtual environments and allows a tangible interface metaphor to be used for object manipulation (Singhal et al., 2012)

Statement of the Problems

General Problem

The thesis is concerned with Interface Computer College - Iloilo Campus having confusing learning materials that sometimes lack further visual demonstration and explanation.

Specific Problem

1. How does the level of interactivity and engagement in the learning materials affect the students?
2. How does the ability of students to visualize the lesson impact their ability to engage in further discussion?
3. How does ineffective communication, particularly the lack of demonstrative examples, impact students' ability to comprehend and master particular subject matters?

Objectives of the Study

General Objective

The main objective of the study is to develop an Augmented Reality application to help students of Interface Computer College - Iloilo Campus to encourage and visualize their lessons.

Specific Objectives

1. To provide engaging and interesting lesson material that students can physically interact with and visualize in the Augmented Reality application.
2. To provide students with a way to enhance their understanding of the lessons with the help of interactable virtual models.
3. To provide teachers and students with an application to be used as a tool to assist in teaching certain topics that are best suitable for Augmented Reality.

Conceptual Framework

Conceptual Diagram

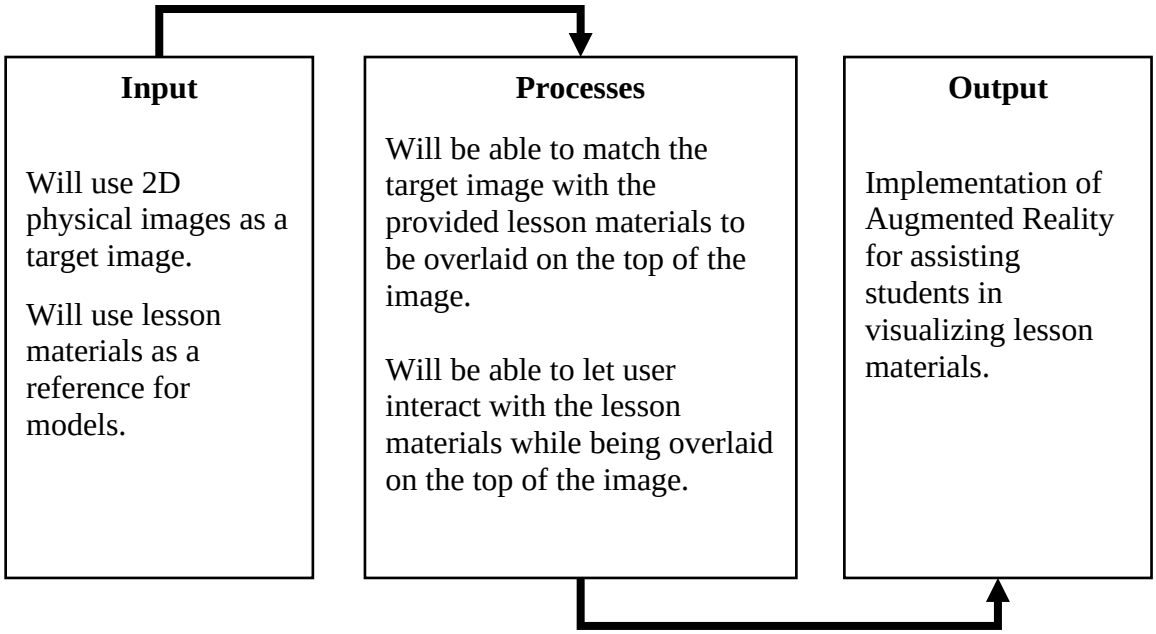


Figure no.1 (Conceptual Framework)

Scope and Limitations

Some topics in school are better understood in a visual manner for students. However, learning materials are often perceived as boring and static. It lacks engagement of the learners and some even do better if they have an overview visualization of the material where they can interact with it. With our AR app, we aim to solve this problem by allowing the student to use their phone to point at a picture of the model, and on the screen, there will be a 3D model so students can interact with it.

The proposed design and development of the Augmented Reality App for assisting learners will only focus on the following features:

- Standalone application
- Only Android OS version 10 and above
- Mobile phones with at least 4GB of RAM
- Mobile phones that have a camera module

Significance of the Study

Generally, the result of this study will contribute to the following beneficiaries:

To the School. The proposed system will help Interface Computer College - Iloilo Campus to have an Augmented Reality application that helps assist students in visualizing their lessons.

To the Teachers. The proposed system will help the teachers enhance their teaching experience by helping students create new skills, and explore new academic interests.

To the Students. The proposed system will help students visualize their lessons.

To the Researchers. The researchers also gain benefits from this study because it enhances the skills and knowledge of each member and enhances the real physical world that is achieved by developing this system.

To the Future Researchers. The proposed system will help future researchers to have a grasp of the abstract concepts created by Augmented Reality, using the interaction and experimentation that Augmented Reality technologies offer.

CHAPTER II
REVIEW OF RELATED LITERATURE AND STUDIES

The purpose of this chapter is to review the literature relevant to this study's thesis problem. The literature review provides an overview of the thesis conducted on the topic of Augmented Reality, both locally and internationally. The review of literature is intended to situate the current study within the larger field of study and to provide a comprehensive understanding of the current state of the thesis on the topic. By reviewing the literature, the researcher is able to identify key concepts, theories, and research findings that are relevant to the topic of this study. The literature review also allows the researcher to identify gaps in the existing literature and highlight how this study will contribute to the existing knowledge in the field.

Foreign

Augmented Reality in Education and Educational Games Implementation and Evaluation: A Focused Literature Review

Studies on augmented reality (AR) in education and in particular AR games in education are gaining impetus worldwide. This area has been actively developing over the past decades relative to the dawn of the 4th industrial revolution and the rapid growth of digital technologies. The present review in the field of educational AR consists of a focused literature review on specific research questions regarding the effective utilization of AR in education and AR gaming applications in the everyday classroom.

Moreover, the review studies knowledge and skills enhancement, teachers' roles, relevant theories, and evaluation techniques.

From the analysis of 78 selected articles, specific conclusions are drawn and a proposition of a series of recommendations and future research in certain areas of educational AR is portrayed. Respectively, several acknowledged issues and limitations regarding the research and the subject area such as infrastructure, curriculum correspondence, AR games in classroom instruction and evaluation as well as the educators' acceptance and contribution.

Augmented Reality in Science Education: An Application in Higher Education

Effective usage of augmented reality (AR) depends on good integration into the learning environment. Based on a qualitative research approach, this study investigates the effects of using AR technologies in science education to support the effective use of AR. Students' experiences of AR were gathered using a prepared questionnaire form. Within the scope of science education, AR was used in a university-level chemistry course. Using theme analysis, descriptive themes were created by analyzing the content of completed questionnaires in written texts. Descriptive expressions obtained from the written text were determined by free coding. These codes were then matched with appropriate themes and illustrated in the form of branched trees. The study results demonstrated that AR is an optimal tool for teaching abstract subjects that do not feature direct observation and examination in science education. Students also have positive opinions about the use of AR in other courses in science education. Another important result from this study revealed that AR software interfaces require improvements to be suitable as teaching material.

In addition, several recommendations have been presented for the better integration of AR into the learning environment.

Augmented Reality and Gamification in Education: A Systematic Literature Review of Research, Applications, and Empirical Studies

This study scrutinizes the existing literature regarding the use of augmented reality and gamification in education to establish its theoretical basis. A systematic literature review following the Preferred Reporting Items for Systematic Reviews and Meta-Analyses (PRISMA) statement was conducted. To provide complete and valid information, all types of related studies for all educational stages and subjects throughout the years were investigated. In total, 670 articles from 5 databases (Scopus, Web of Science, Google Scholar, IEEE, and ERIC) were examined. Based on the results, using augmented reality and gamification in education can yield several benefits for students, assist educators, improve the educational process, and facilitate the transition toward technology-enhanced learning when used in a student-centered manner, following proper educational approaches and strategies and taking students' knowledge, interests, unique characteristics, and personality traits into consideration.

Students demonstrated positive behavioral, attitudinal, and psychological changes and increased engagement, motivation, active participation, knowledge acquisition, focus, curiosity, interest, enjoyment, academic performance, and learning outcomes. Teachers also assessed them positively. Virtual rewards were crucial for improving learning motivation. The need to develop appropriate validation tools, design techniques, and theories was apparent.

Finally, their potential to create collaborative and personalized learning experiences and to promote and enhance students' cognitive and social–emotional development was evident.

A Qualitative Case Study in Augmented Reality Applications in Education: Dimensions of Strategic Implementation

The main purpose behind technology integration is to improve the teaching and learning process. Augmented reality (AR) is a new instructional tool in the educational field. Current literature shows AR integration is successful in the United States. However, it does not exist in Kuwait. Despite the time and money invested to integrate different instructional tools, teachers in Kuwait have been unaware of the existence of the AR application and its potential use in the classroom. There is a need to help teacher educators in Kuwait use AR applications and maximize the benefits of this technology for students' best interests. This study explored the use of AR among teacher educators in the United States who currently used this technology in their classrooms. It also explored the opinions of Kuwait veteran teachers regarding AR technology integration, designed a process for strategic implementation aimed at teachers in Kuwait, and acted as a guide to follow for AR integration in education in Kuwait. The target population consisted of teacher educators from the United States who have integrated AR applications into their teaching and veteran public elementary school teachers from Kuwait. This study employed a qualitative case study design. A number of interviews and focus group discussions were conducted to collect data. Major themes were developed for both samples. U.S. iv participants shared their educational experiences with AR integration.

Although teacher participants from Kuwait were initially overwhelmed with AR, this study found they would be willing to implement AR if supported by the government.

The findings of the study provided a clear view of how to integrate AR technology as an educational tool to vary and improve instructional tools in Kuwait classrooms to meet young learners' needs and interests. The recommendations for strategic implementation were specifically tailored for public schools in Kuwait. A suggestion for further research was to have a second phase of the study that would examine teacher participants' experiences with AR integration. Finally, the implications of this research study supported every educator who had never considered nor integrated AR as an instructional tool, specifically teachers in Kuwait.

Augmented Reality Use, Application Use, and Application of Augmented Reality in General

Augmented reality technology is a developing tool in the field of promising technologies. Research showed AR is a growing technology that is popular in each field it serves. "It continues to develop over the decades and works its way into the modern 25 technological landscape of today" (Kipper & Rampolla, 2013, p. 14). Augmented reality technology can be used in advertising, navigation, sightseeing, entertainment, games, and education. According to Lee (2012), "Augmented reality has been put to use by a number of major companies for visualization, training, and other purposes" (Lee, 2012, p. 403). As mentioned previously, many big companies have integrated AR to advertise their products and provide their customers with the best services.

Navigation

Augmented reality is useful for navigation as it provides interesting information. For instance, Yelp and NRU applications mainly help users find places to eat, drink, and shop with real-time visual directions to the places (Kipper & Rampolla, 2013).

Sightseeing AR technology has the ability to enhance the user's sightseeing experience and explore unique details of a place, event, or character by displaying interesting digital information and animations in real-time. For instance, New York's Museum of Modern Art hosted an exhibit in 2010 (Kipper & Rampolla, 2013). The museum applied AR technology so visitors were able to view hidden exhibitions by using a certain AR application for iPhones or Android phones. A couple of specifically designed AR applications are currently on the market for tourism.

Entertainment and Games

The entertainment industry generates billions of dollars each year (Kipper & Rampolla, 2013). Producers along with entertainers are working to provide a better experience for audiences and go beyond their expectations. With the continued growth of new technology, the definition of entertainment for audiences is being challenged. Augmented reality applications have tremendous potential to provide a unique 26 entertainment experience that allows users to interact with entertainment elements. Currently, AR-enabled games are now available for both mobile devices and desktop computers.

Education

Technology is changing education, especially with the presence of mobile and tablet devices that access the Internet. The use of different types of technology in the educational setting could simplify complex information, motivate learners, and engage them in the learning process. For example, learners interact with the digital smart board more than the traditional chalkboard. Augmented reality technology is able to create a very interactive learning environment that is appealing to learners as they take control of their own learning and interact with digital objects in a real environment.

Application as an Instructional Tool

Augmented reality is an advanced instructional tool to be used in the classroom. A number of researchers have connected the use of AR technology with student motivation. According to Lee (2012), AR has the ability to strengthen students' motivation and enhance their educational realism-based practices. Balkun (2011) emphasized the idea that "students not only must have access to digital media but also learn how to use technology thoughtfully, creatively, and cooperatively".

Application in Educational Settings

It is important to apply the best and latest learning strategies and tools to support learning and engage students.

"The goal is to provide an enriched experience that gives students access to information and materials not readily available elsewhere" (Balkun, 2011, p. 15). The use of audio and video has a long history in education (University of 27 Guelph, 2006); AR carries this further by allowing interaction with audio and video.

Integration of AR can include learners of all ages. According to those whose research is in the field of AR and education; it is important for every teacher to read and think critically about the value of integrating AR in their curriculum. One of the goals of this research was to see if this was true. It was imperative to understand how AR could be connected to the primary goals and objectives of the content. These connections must be related and able to enhance meaning-making. The connections might look like a puzzle in terms of integrating AR into the curriculum but once the puzzle has been solved, the outcomes are quite impressive.

Advantages of Augmented Reality

A number of strengths and weaknesses are associated with composing and integrating media in the classroom such as intellectual property, copyright, class attendance, production support, software and hardware issues, accessibility, complexity, and learning styles. In addition, “there is no doubt that learning to use digital media 32 effectively requires a commitment of both time and resources” (Balkun, 2011, p. 21). Some studies have demonstrated that a large commitment is not the case anymore (Journet, 2007; Vogt & Shingles, 2013) since the use of these applications is mostly through smartphones and tablets and most teachers along with learners are very familiar with these devices. Particularly, mobile devices have increased the number of AR tools and applications. Added to that,

AR tools and applications do not require specific knowledge or expertise to be implemented. “Complexity of AR is being taken away from the end-user product, and is gradually less likely to slow down or stop the expansion of AR” (Vogt & Shingles, 2013, p. 56).

Consequently, the complexity associated with digital media applications and tools in the field of education is not the biggest limitation and will not affect possible expansions of AR into many educational settings.

An overview of research literature showed that learners benefit from learning if it meets with students' needs to expand and grow their knowledge. Sansone (2014) stated, "Students have the opportunity to become an active part of their learning" (p. 73). Accessibility in terms of Internet connections or access to computers needs to be considered. However, since most digital media applications and tools are integrated in mobile devices associated with wireless internet and Wi-Fi features, an expensive overlay of wired connections is not necessary.

One of the advantages of AR technology is "students generally know more about the mobile applications and are certainly more immersed in sound and in the visual, and are more fluent in new electronic genres" (Journet, 2007, p. 116). Teachers can benefit from students' backgrounds and try to merge the use of AR with the academic context. Scholz (as cited in (Balkun, 2011) supported the use of technology as he stated, [It] "can 33 help learners to become more active participants in public life and, moreover, radical pedagogy and civic engagement" (p. 18).

Teachers cannot create certain experiments in the classroom, bring places such as the Galaxy to the classroom, or show students how human bodies look internally. In contrast, augmented reality applications and tools provide an alternative solution to creating unique learning experiences for learners. The evolution of AR offers exciting and engaging opportunities to enhance educational purposes and facilitate learning.

Local

ARGeo Philippines: A Mobile Augmented Reality for Learning Philippine Geography

The perceived difficulties and challenges in teaching geography compel the use of innovative instructional approaches. Hence, this study developed an augmented reality (AR) application for m-learning, 'ARGeo Philippines', to provide undergraduate-level students with an interactive learning experience of Philippine geography. We applied the waterfall model in software development and subjected the AR application to software quality tests, usability tests, and pedagogical effectiveness tests. Using the SCORM standard, software quality requirements for the AR application were met generally. For usability as a mobile application, the AR application was rated overall as 'highly' acceptable (Mean=4.15, SD= 0.29) using the MLUAT instrument. Pedagogical potential still requires further verification due to the statistically insignificant result; however, using the AR application resulted in higher learning gains than those from the traditional learning approach. Thus, it might be beneficial to further explore the use of the AR application as an alternative learning material for the academe.

A Review of Augmented Reality Apps for an AR-Based STEM Education Framework

Within the past two decades, Augmented Reality (AR) applications have received increased attention. Augmented Reality is now widely used in the education sector at level K-12. AR is expected to be generally adopted in two-to-three years in higher education and four-to-five years in K-12. Applying AR technology in the education sector, especially in STEM subjects, can result in having a smart campus.

In adopting a SMART Campus strategy, education practitioners must address many intrinsic issues in science, technology, engineering, and mathematics (STEM) research. For example, in physics, there are expensive or insufficient laboratory systems, system faults, and difficulty simulating other experimental circumstances; in technology, many schools do not have enough computers; in engineering, there are only a few instructors who are knowledgeable in computer-aided design (CAD); and in mathematics, few teachers incorporate technology into their lessons often because they believe it is still better to teach through the traditional methods. Hence, in this paper, we discuss how AR is being used now in different learning areas in STEM to open new doors to researchers and teachers as they transition their schools into SMART campuses with the use of AR apps. Aligned with this, a suggested framework for school administrators and policymakers is proposed based on a review of the positive benefits of different AR apps.

Developing an Educational Augmented Reality Game on the Battle of Mactan Using the Intel Perceptual Computing Kit

The purpose of this study is to produce an educational augmented reality game for elementary students taking up Philippine history in their Araling Panlipunan class. The product aims to address the lack of educational software on Philippine-specific topics by making use of the advantages offered by augmented reality. To achieve this goal, the research group developed an educational game on the Battle of Mactan using the Unity game engine in conjunction with the Intel Perceptual Computing Kit. Users play as a contestant in a game show called Zubu: Battle of Mactan, competing in a quiz show with AI opponents and playing mini-games to learn about the historical battle.

Information was organized into three segments: people, events, and culture. Augmented reality features such as gesture recognition and hand tracking were used

to facilitate the controls of the game, alongside traditional keyboard and mouse input. The game showed promising short-term results through the demonstrated improvement between pre-game and post-game test scores of the students. The game may prove to be a helpful tool in learning but may still need further development in order to fully maximize its potential as an aid in teaching.

MAGIS: mobile augmented-reality games for instructional support

We present Mobile Augmented-Reality Games for Instructional Support (MAGIS), a framework for the development of mobile augmented reality (AR) games for education.

The framework supports off-the-shelf, state-of-the-art technologies that enable AR tracking and rendering on consumer-level mobile devices, and integrates these technologies with content-generation tools that simplify the development of educational AR games, especially those that extensively use narrative-based game design and player-location tracking such as location-based historical or museum adventure games. We use *Igpaw: Intramuros*, a proof-of-concept game developed using MAGIS, to help describe the current state of the framework and to show its efficacy for implementing outdoor location-based educational games, and we briefly outline future development plans to improve MAGIS' AR support (especially those involving indoor scenarios). As well as to improve the instructional design and authoring phases of AR applications written using this framework. Transcending the Philippine Educational Culture through Virtual and Augmented Reality with the Integration of the Learning Management Systems.

Amidst the current pandemic situation right now, the Department of Education and the Commission on Higher Education are facing a lot of challenges that will take place this coming opening of classes, and part of their initiative is to adopt the mode of flexible learning. In the case of other Higher Education and other educational sectors which cater to practical and situation cases either scientific or technical approaches in facing the real-time situation. The Researcher is proposing a new alternative flexible learning that will ease up the situations of the current pandemic at all education levels, which is the integration of Virtual Reality, AR learning content as part of the educational authoring tools, a web-based platform for the development of Augmented content, the distribution of which is accomplished through standardized Learning Management Systems (LMS),

Using SCORM packages library, Furthermore, a collaborative framework solves the problem of standard integration of Augmented and Virtual Reality applications in education offering a distributed framework which is the Learning Management System platform, (Barbadillo, Barnera, Goñi, & Sánchez, 2014). Upon the implementation of these interactive and immersive technology platforms, higher-order thinking or critical thinking skills will develop further.

Transcending the Philippine Educational Culture through Virtual and Augmented Reality with the Integration of the Learning Management Systems

Amidst the current pandemic situation right now, the Department of Education and the Commission on Higher Education are facing a lot of challenges that will take place this coming opening of classes, and part of their initiative is to adopt the mode of flexible learning.

In the case of other Higher Education and other educational sectors which cater to practical and situation cases either scientific or technical approaches in facing the real-time situation. The Researcher is proposing a new alternative flexible learning that will ease up the situations of the current pandemic at all education levels, which is the integration of Virtual Reality, and AR learning content as part of the educational authoring tools, a web-based platform for the development of Augmented content, the distribution of which is accomplished through standardized Learning Management Systems (LMS) using SCORM packages library,

Furthermore, a collaborative framework solves the problem of standard integration of Augmented and Virtual Reality applications in education offering a distributed framework which is the Learning Management System platform, (Barbadillo, Barnera, Goñi, & Sánchez, 2014). Upon the implementation of these interactive and immersive technology platforms, higher-order thinking or critical thinking skills will develop further.

CHAPTER III
RESEARCH DESIGN AND METHODOLOGY

This chapter provides an overview of software development methodologies and the technical background of the thesis. It discusses hardware and software requirements, costs, benefits, data-gathering techniques, and statistical analysis. The chapter covers system design, including the agile software development methodology. It also explores database design and management systems. Overall, this chapter is essential to the thesis development and lays the groundwork for subsequent chapters.

Software Development Methodologies

Technical Background

A. Hardware Requirements

Developer	User
RAM: 4GB DDR4	Camera Module
STORAGE: 250GB SSD	Internal memory min 1GB
	CPU
	GPU

Table no.1 (Hardware Requirements)

A. Software Requirements

Software	Description
Operating System	Android 10 and above

Table no.2 (Software Requirements)

A. Costs and Benefits

Hardware	Price
Sony VAIO E Series 14 P Intel ® Core i5-3210M CPU @ 2.5GHz 4 CPU logical cores 8GB RAM Windows 10 64Bit OS	₱23,000.00
Printer Epson L120	₱6,900.00
Software	
Unity Game Engine Vuforia SDK Blender	FREE

Audacity Visual Studio Code	
Adobe Photoshop CS6	₱5,795.25
Ink	₱1,040.00
Bond Paper	₱277.00
Grammarian	₱1,500.00
Book Bind	₱1,800.00
Hardware Cost	₱29,900.00
Software Cost	₱5,795.25
Other Cost	₱4,617.00
Total Cost	₱40,312.25

Table no.3 (Cost and Benefits)

Data Gathering Techniques

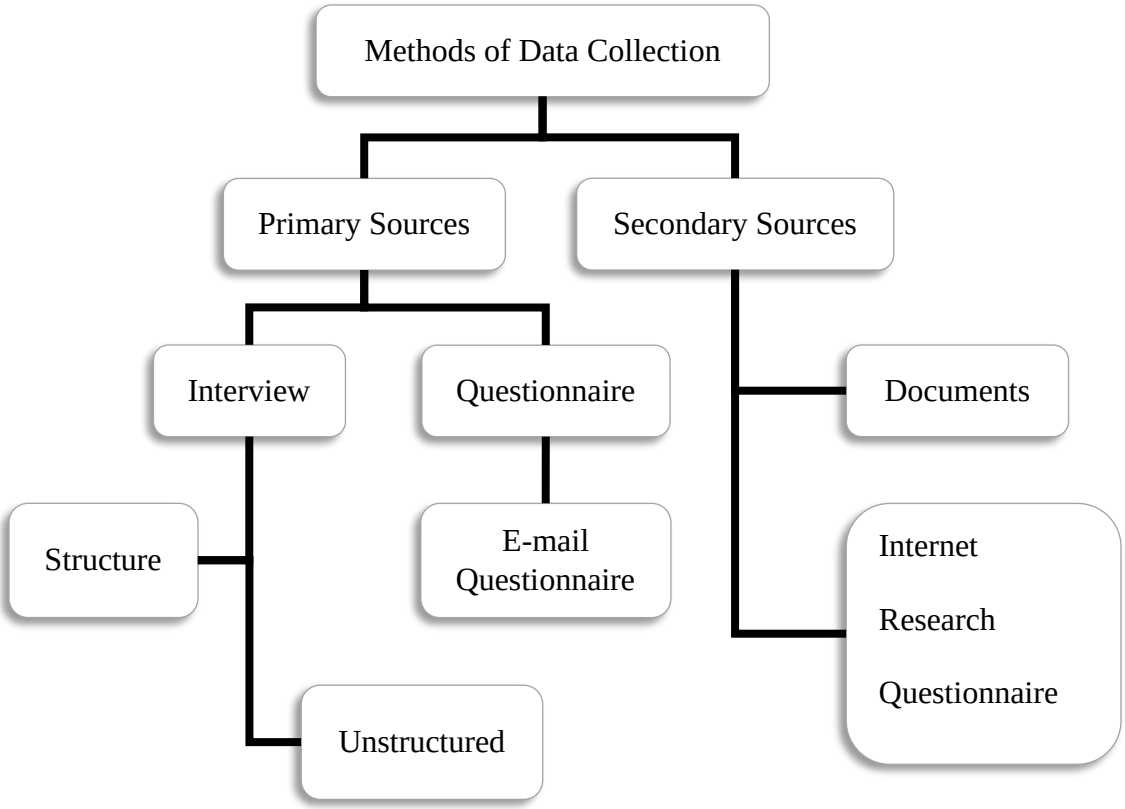


Figure no.2 (Data Gathering Techniques)

Interview. The researchers conducted an interview to get users' opinions and experiences about the app. Interviews can give more detailed information, allowing researchers to identify what users liked and disliked about the app. This information can be used to improve the app and design a better AR app.

Survey. The researchers used a survey to collect data from users about their experiences and opinions of the app.

Questionnaire. The researchers used a questionnaire to gather information from app users. Researchers design questionnaires to collect user data. The questionnaire is useful because it helps researchers understand the users. It also helps researchers learn how users use the app and determine their thoughts.

Research Method

The approach to this study is qualitative in that the researchers collected data from interviews and questionnaires conducted by them. All results were reviewed thoroughly to come up with answers as to how the app can be improved to make it a better app. In addition, we added meaningful and useful topics.

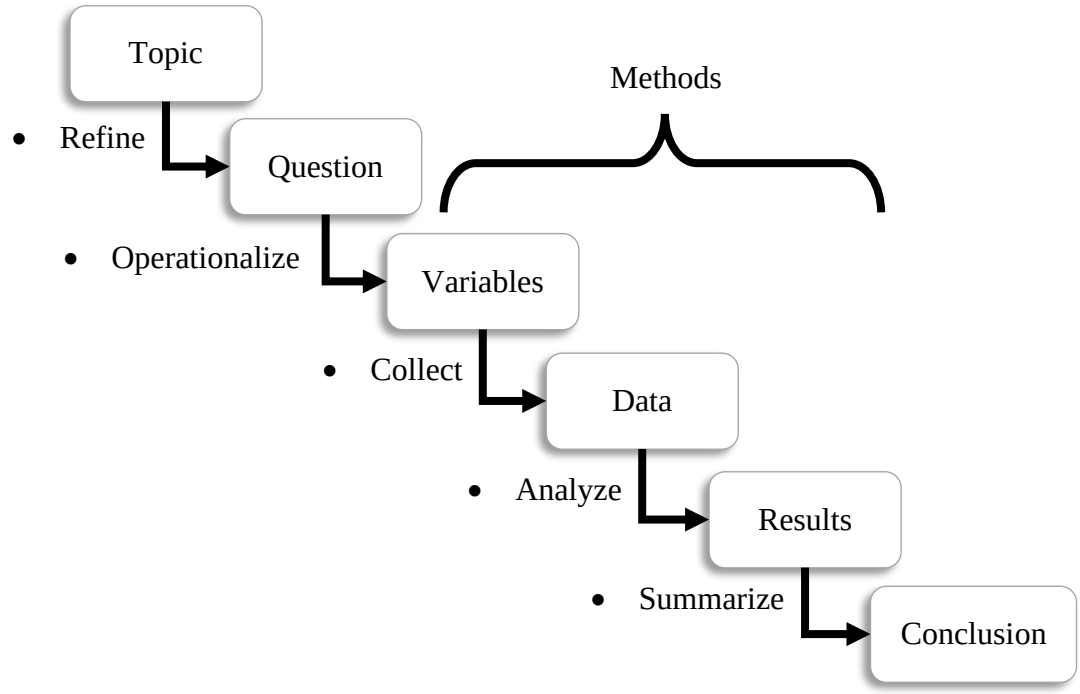


Figure no.3 (Research Design)

Software Model

Requirements Analysis. During this phase, the researchers discussed when and how the app would work together, including the hardware, software requirements, and the possible models and systems needed in the AR app.

Design Prototyping. The goal of this phase was to gather more information to design and develop the AR app in an efficient way.

Customer Evaluation. During this phase, the researchers conduct demos and surveys to present the AR app features and capabilities to the customer for additional feedback and evaluation.

Reviews and Update. During this phase, the researchers would review the feedback received and would update the AR app on what would best result in positive feedback.

Development. At this phase, the researcher would develop the AR app as a whole and finalize the project taking into account the revisions and feedback from the previous phases.

Testing. As part of this phase, the researchers conducted several tests to ensure that the AR app created was capable of achieving the study's goals. Also, the researchers tested whether the functions worked properly.

Maintenance. During this phase continues maintenance to the AR app would be provided to ensure

Prototyping Model

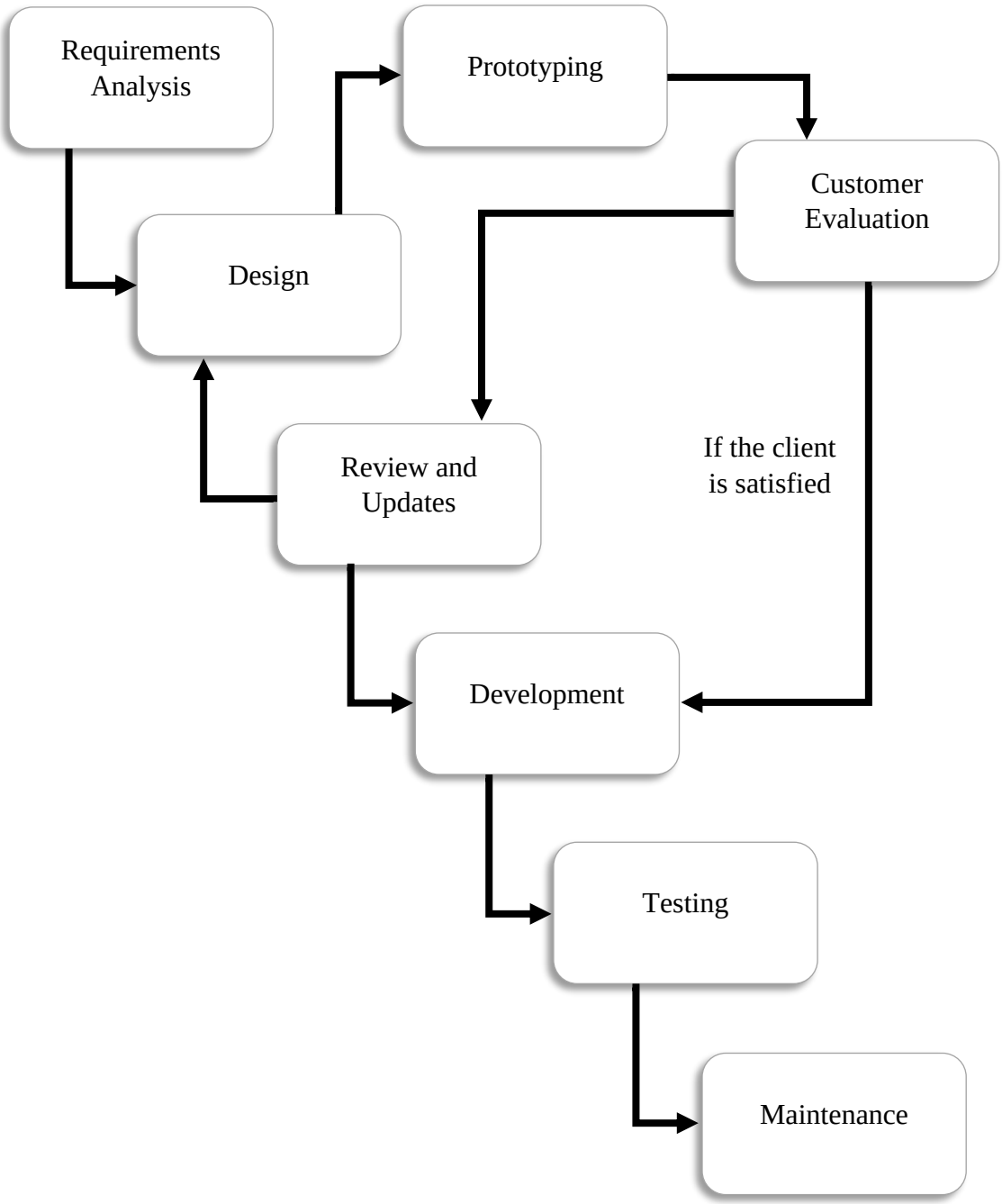


Figure no.4 (Prototyping Model)

Database Structure

Database Management System

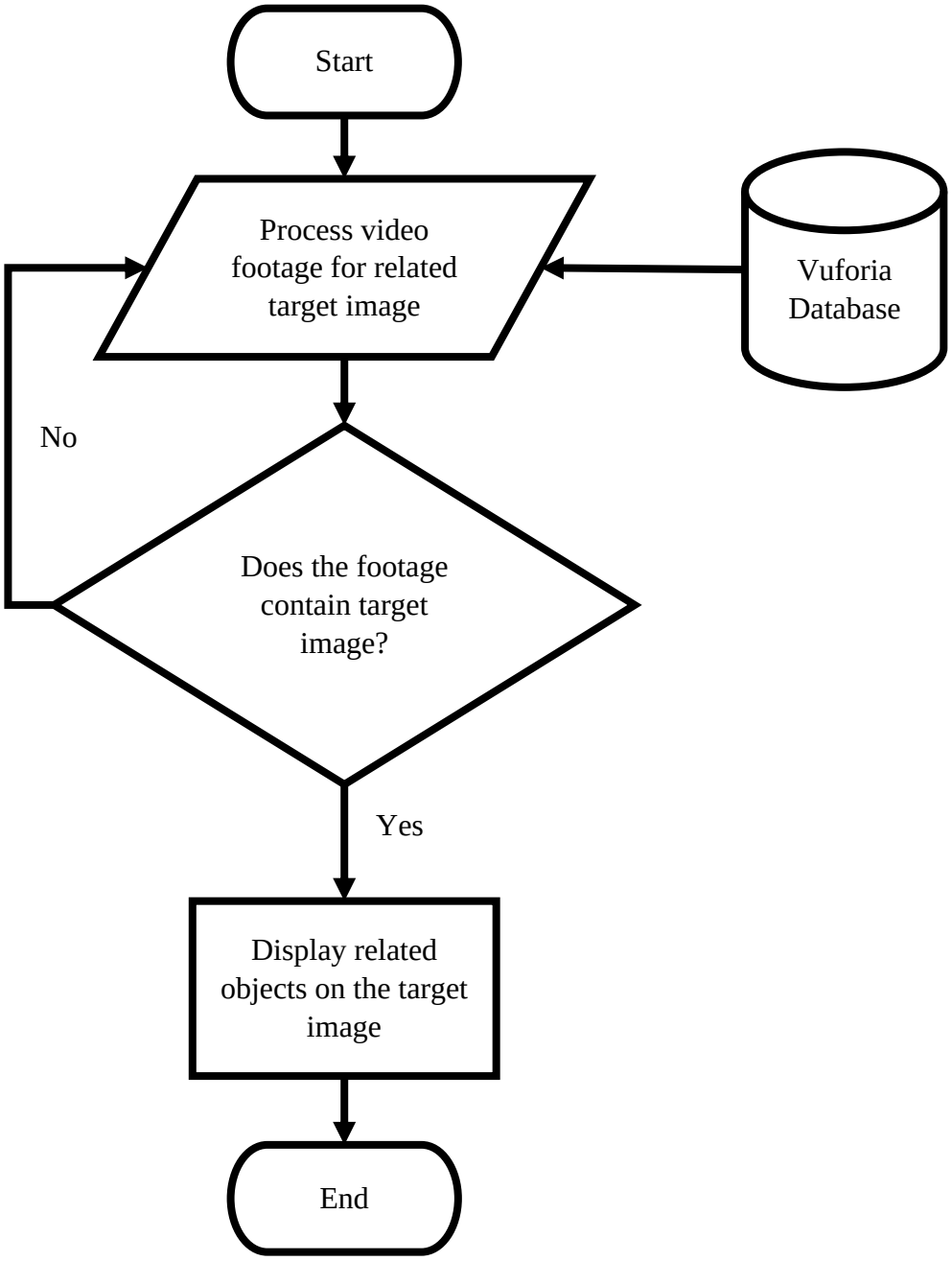


Figure no.5 (Database Management System)

CHAPTER IV
PRESENTATION, ANALYSIS, AND INTERPRETATION

System Overview

This chapter introduces the main design issues and needs that concern the implementation of the “Use of Augmented Reality in Education for Assisting Students of Interface Computer College Iloilo, and its related features.

The proposed system was developed using C# programming language by the use of Unity Game Engine and Vuforia SDK. The system improved by assisting students with their computer-related lessons.

The proposed system was intuitive, and interactive and had a user-friendly Graphical Interface that even those with limited knowledge of the use of mobile phones could easily operate.

The system was designed to support students in visualizing computer-related topics.

System Objectives

- The application will have the capacity to be intractable.
- The application will overlay 3D models, images, and video.
- The application will have the ability to capture raw video footage from the mobile phone's camera.
- The program will be able to present computer-related topics in an AR form.

Data Gathering Results

Background of the Company

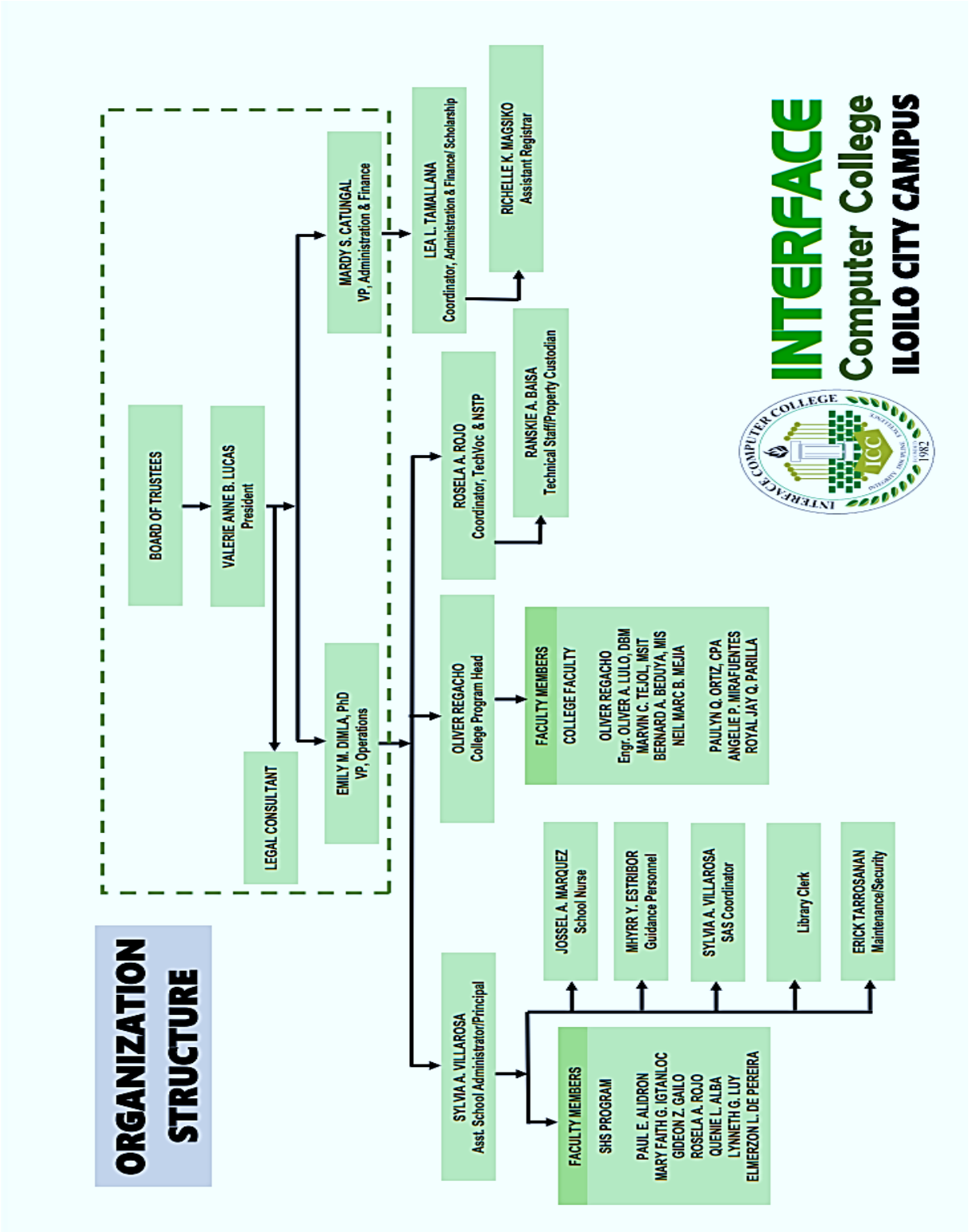
Interface Computer College then known as Interface Computer Learning Center was established in 1982 on the 2nd floor of the Cartimar building along Claro M. Recto Avenue in Quiapo Manila. It then became a college in 1994 when IT-related programs were added to its already existing wide-range portfolio of short-term courses. Proud to say that all the programs and courses offered are duly recognized and authorized by the Commission on Higher Education (CHED)and Technical Skills and Development Authority (OWWA), the Civil Service Commission (CSC), and the Department of Science and Technology (DOST).

Responding to the clamor and growing demands of the public to be more accessible, ICC opened its door to reach out to more people thereby opening branches in Caloocan, Cebu, Davao, Iloilo, and in Cabanatuan. Since then, ICC has grown and pioneered to become one of the largest private non-sectarian institutions in the Philippines. It boasts of its status as a non-franchised institution with the end view of sustaining its standard of quality education.

Committed to improving the curricula offering, Interface Computer College has taken a big leap in offering virtual classroom through e-learning where students can interact with the instructions through the internet, e.g., download course topics, submit assignments, know their standing grades, and even takes examination, all through the use of the internet.

Today, with thousand and more students added to the alumni every year, ICC takes pride in the pursuit of academic and technical excellence in consistently participating and contributing to the technological progress locally and globally. Across the myriad socioeconomic and political landscapes of the country, one thing remains constant with ICC- the vision of committing itself to enhance the potential skills of the students through the delivery of quality, equitable, accessible, and yet affordable education to all. Its mission is “To produce graduates equipped with the competence, excellence, and character necessary to achieve a transformative impact on society”. And vision “We are the premiere learning institution providing holistic and innovative education that empowers our students to be globally competitive and responsible members of society”.

Organizational Chart



Demographic Profile of the Respondents

This section provides an overview of the respondents in this study. The respondents consisted of teachers and students in Interface Computer College Inc. The study included 4 teachers, of whom 50% were male and 50% were female, and 7 students, 28.6% were male and 71.4% were female. The data were collected through surveys and interviews to obtain information on various factors related to the use of the AR ICC Visualizer app for the teaching and learning of teachers and students.

	Amount	**Percentage**
GENDER		
MALE	**2**	**50%**
FEMALE	**2**	**50%**
TOTAL	**4**	**100%**
HANDLED SUBJECTS		
SOFTWARE	**3**	**75%**
HARDWARE	**1**	**25%**
TOTAL	**4**	**100%**

Table no.4 (Teachers)

Based on the table, our female respondents are 2 (50%) while our male respondents are 2 which is (50%). Since our survey has 50%-50% males and females, most respondents are males and females. Also, the majority of our respondents handled software subjects since it has a percentage of 75%.

	Amount	Percentage
GENDER		
MALE	**2**	**28.6%**
FEMALE	**5**	**71.4%**
TOTAL	**7**	**100%**
COURSE		
SHS TVL/ICT	**0**	**0%**
BSCS	**0**	**0%**
BSIT	**7**	**100%**
BSCoE	**0**	**0%**
TOTAL	**7**	**100%**
YEAR		
SHS GRADE-11	**0**	**0%**
SHS GRADE-12	**0**	**0%**
1ST YEAR	**0**	**0%**
2ND YEAR	**0**	**0%**

3RD YEAR	**2**	**28.6%**
4TH YEAR	**5**	**71.4%**
TOTAL	**7**	**100%**

Table no. 5 (Students)

According to the student table, 71.4% of the respondents are female while 28.6% are male. Therefore, most of our respondents are female. In terms of course and year, the majority of our respondents are BSIT students because it has 100% students. The majority of respondents are in their fourth year, with 71.4% of respondents.

CHAPTER V
CONCLUSIONS AND RECOMMENDATIONS

This chapter presents the conclusions and recommendations derived from the augmented reality (AR) study in education at Interface Computer College - Iloilo Campus. It provides a brief overview of the research objectives and methodology. Additionally, practical recommendations are offered to guide the students in effectively utilizing AR to enhance the learning experience and support them.

The recommendations address AR integration into the curriculum. They include providing technical support and resources, training for educators, fostering collaboration, evaluating impact, establishing partnerships, and staying updated with AR advancements. By following these recommendations, Interface Computer College - Iloilo Campus can create an engaging and innovative learning environment that benefits students.

This chapter aims to provide a concise roadmap for students in order to gain a deeper understanding of AR technology. By implementing these recommendations, Interface Computer College - Iloilo Campus can enhance student engagement, provide personalized learning experiences, and cultivate essential skills in computer science and information technology.

Conclusions

In conclusion, this study explored the use of augmented reality (AR) in education to assist Interface Computer College Iloilo students.

Through an in-depth examination of literature and empirical evidence, AR holds significant potential for transforming Interface Computer College - Iloilo Campus students' learning experience.

This study indicates that AR can enhance student engagement and motivation by providing immersive and interactive learning environments. By overlaying virtual elements in the real world, AR enables students to visualize complex concepts, manipulate virtual objects, and engage in hands-on activities, leading to a deeper understanding and retention of knowledge.

Moreover, AR facilitates personalized and adaptive learning, catering to Interface Computer College - Iloilo Campus students' unique needs and learning styles. It offers interactive simulations, virtual experiments, and real-time feedback, empowering students to actively participate in their own education and take ownership of their learning journey.

Furthermore, AR promotes collaboration and social interaction among students. By engaging in shared AR experiences, students can collaborate on projects, solve problems collaboratively, and participate in cooperative learning activities. This fosters the development of essential skills such as teamwork, communication, and critical thinking. These skills are highly valued in computer science and information technology.

However, it is important to acknowledge the potential challenges and limitations associated with AR implementation in the context of Interface Computer College - Iloilo Campus. Technical considerations, such as device compatibility and infrastructure requirements, need to be carefully addressed to ensure seamless integration.

Additionally, adequate training and support for educators are crucial for effectively incorporating AR into the curriculum and maximizing its benefits for students.

In conclusion, the integration of augmented reality in education holds great promise for assisting students at Interface Computer College - Iloilo Campus. By providing immersive and interactive learning experiences, personalizing instruction, fostering collaboration, and enhancing student engagement, AR has the potential to revolutionize the educational experience at the institution. Future research and development in this area are necessary to overcome these challenges, tailor AR applications to the specific needs of Interface Computer College - Iloilo Campus, and further explore the pedagogical implications of AR, ultimately paving the way for a more innovative and effective learning environment for students.

Recommendations

- **Integration of AR into the Curriculum.** Interface Computer College - Iloilo Campus should consider integrating AR technology into the curriculum across various subjects and courses. This can be achieved by identifying key learning outcomes that can be enhanced through AR experiences and designing appropriate AR-based learning activities.
- **Technical Infrastructure and Resources.** The students should invest in the necessary technical infrastructure, such as compatible devices and reliable network connectivity, to support AR implementation in classrooms. Additionally, providing access to AR resources, applications, and software tools can enable both educators and students to effectively utilize AR technology.

- **Professional Development for Educators.** It is crucial to provide training and professional development opportunities for educators at Interface Computer College - Iloilo Campus to familiarize them with AR technology and its pedagogical applications. This training should focus on instructional strategies, best practices, and the effective integration of AR into teaching methodologies.

- **Research and Evaluation.** Continual research and evaluation should be conducted to assess the impact and effectiveness of AR in assisting students at Interface Computer College - Iloilo Campus. This can involve collecting feedback from students and educators, analyzing learning outcomes, and identifying areas for improvement and further development.

- **Continuous Improvement and Upgrades.** Given the rapid advancements in AR technology, Interface Computer College - Iloilo Campus should remain updated with the latest trends, tools, and applications. Regularly evaluating and upgrading AR hardware and software can ensure that students have access to state-of-the-art AR experiences, maximizing AR's educational benefits and assisting their learning process.

By implementing these recommendations, Interface Computer College - Iloilo Campus can harness the full potential of augmented reality technology to enhance student engagement, facilitate personalized learning, and foster a dynamic and immersive educational environment.

REFERENCES

https://www.researchgate.net/publication/281336331_A_Review_of_Research_on_Augmented_Reality_in_Education_Advantages_and_Applications

Local Literature and Studies

- https://arxiv.org/ftp/arxiv/papers/2203/2203.07024.pdf
- https://alls.ateneo.edu/wp-content/uploads/2014/12/Paper-05-FINAL.pdf
- http://conf.e-jikei.org/ISTSS/2019/proceedings/materials/proc_files/GS/ISTSS2019_GS_A002/CameraReady_ISTSS2019_GS_A002.pdf
- https://www.tandfonline.com/doi/abs/10.1080/10494820.2018.1504305
- https://www.google.com/url?client=internal-element-cse&cx=001431978847466539083:xsldadcvvvo&q=http://ijmaberjournal.org/index.php/ijmaber/article/download/13/3&sa=U&ved=2ahUKEwjOge7C_bH7AhVZrlYBHZ5mCqMQFnoECAIQAg&usg=AOvVaw3WuJuew5BWMn44d_KHUwDF

Foreign Literature and Studies

- https://www.computersandchildren.com/download/augmented-reality-in-education-and-educational-games-implementation-and-evaluation-a-focused-11925.pdf
- https://files.eric.ed.gov/fulltext/EJ1300883.pdf
- https://www.mdpi.com/2076-3417/12/13/6809
- https://digscholarship.unco.edu/cgi/viewcontent.cgi?article=1477&context=dissertations
- https://www.techtarget.com/whatis/definition/augmented-reality-AR

APPENDICES

Appendix A

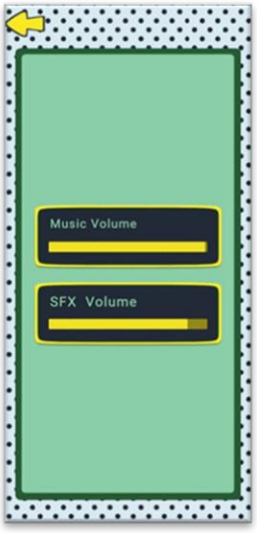
System Design

Home screen

The First Panel the user will see when opening the app. The panel contains the Topics, Settings, About us, Help and Quit Buttons.

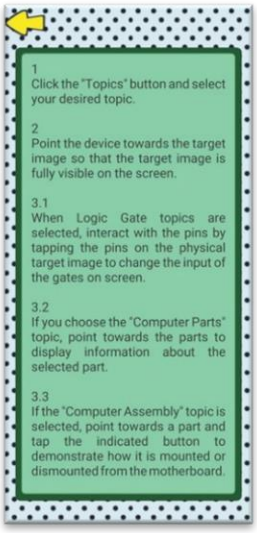
Topic Holder Screen

When the user presses the topics button, the Topic Holder panel will appear where the user will then be able to select the topics to then present.



Settings Panel

When the user presses the settings button, the Settings panel will then appear where the user will then be able to adjust the background and sound effects volume.

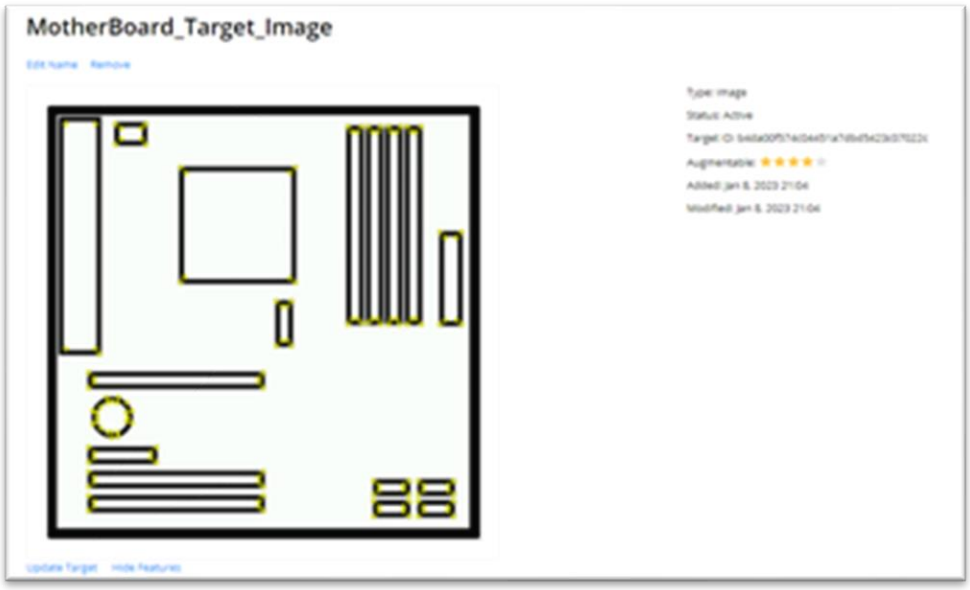


Help Screen

When the user presses the Help button, the Help panel will then appear where the user will be presented with instruction on how to use the application.

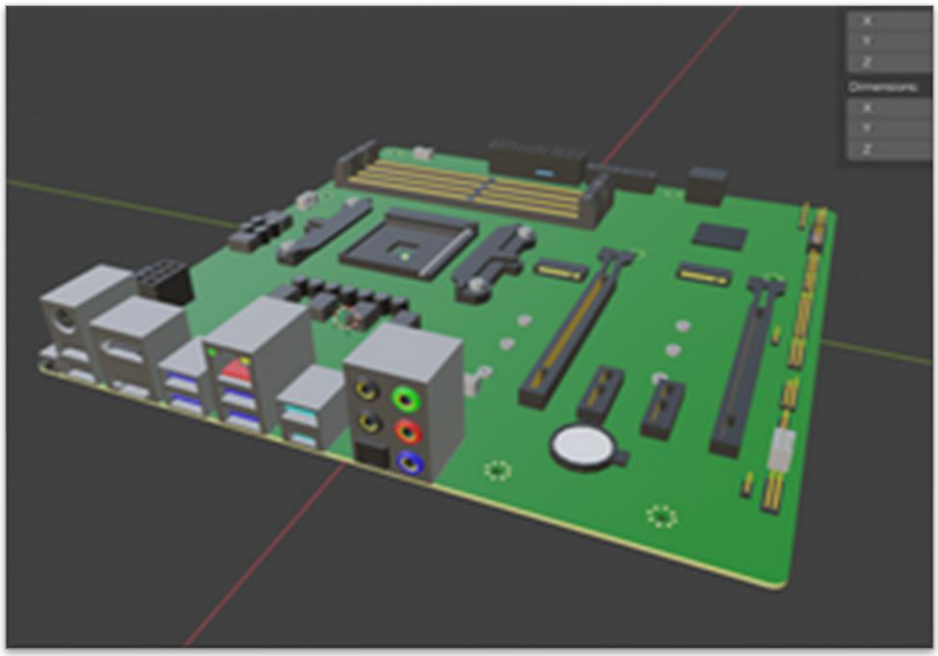
About Us Screen

When the user presses the About Us button, the About US panel will then appear where the user will then be presented with the information about the creators of the application and the purpose of the application



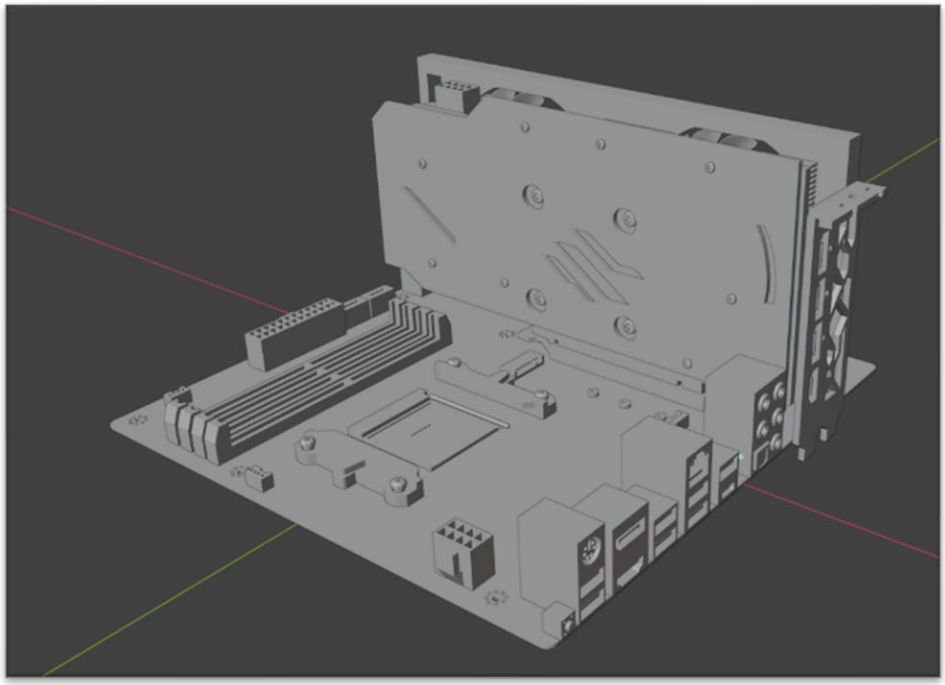
Motherboard Target Image in Vuforia

The image that is stored in the Vuforia Database that will be used in the application to find in the video feed.



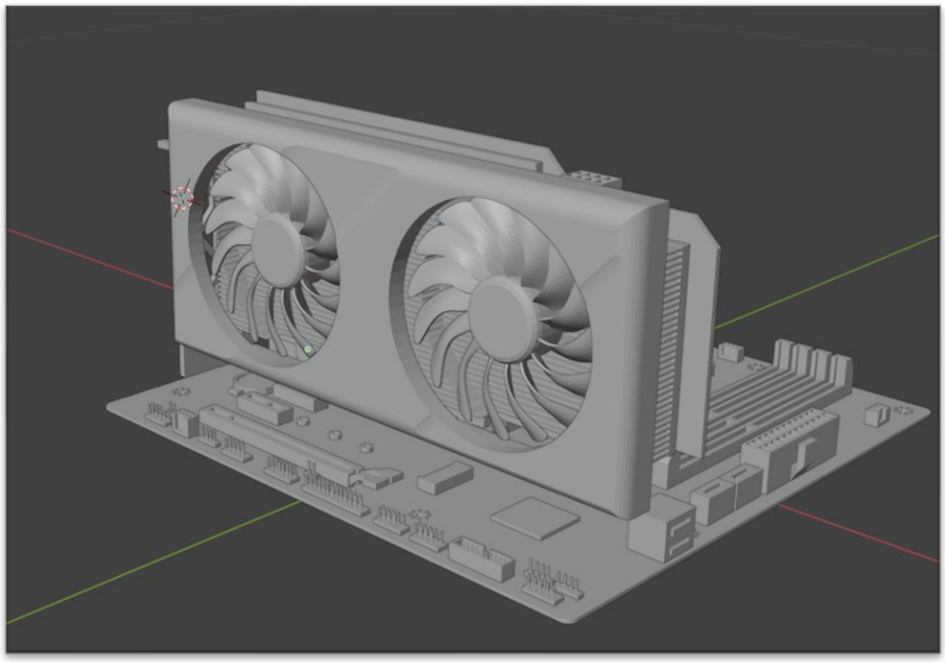
3D Model of the Motherboard in Blender with colored materials

The port side view of the motherboard model with the colored materials applied to provide the user with contrasting of the parts and components of the motherboard



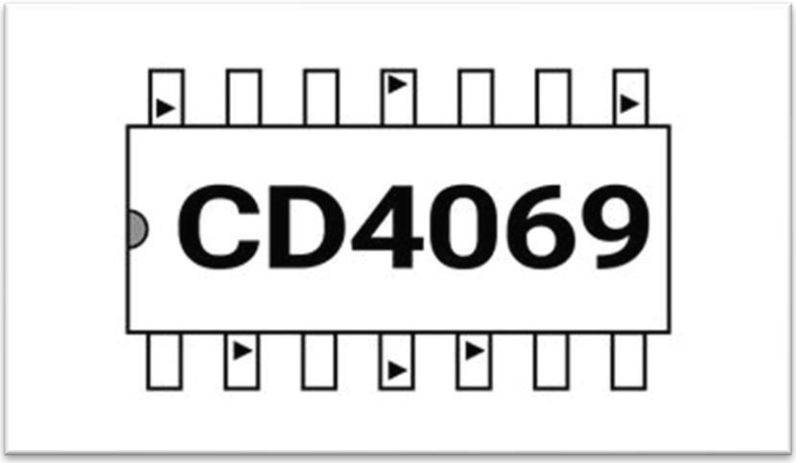
Rear view of the 3D Model of the Motherboard in Blender

The raw 3D model of the motherboard with the GPU equipped



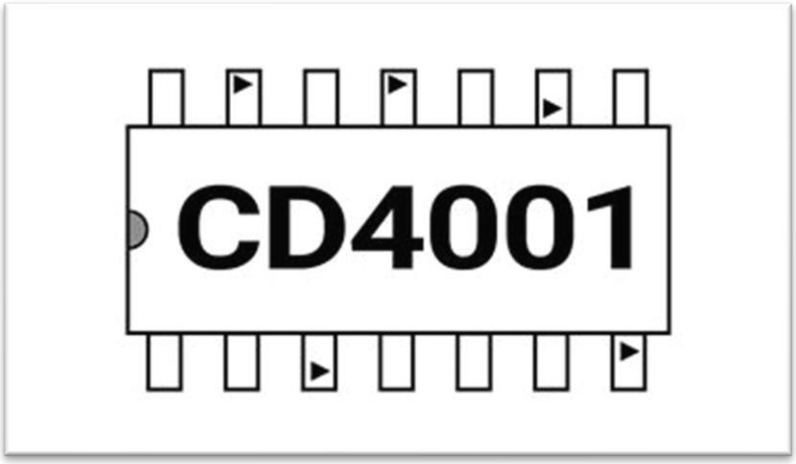
Front view of the 3D Model of the Motherboard in Blender

The raw 3D model of the motherboard with the GPU equipped



IC 4069 NOT Gate Target Image

The target image for the IC 4069 topic to show the 3D model



IC 4001 NOR Gate Target Image

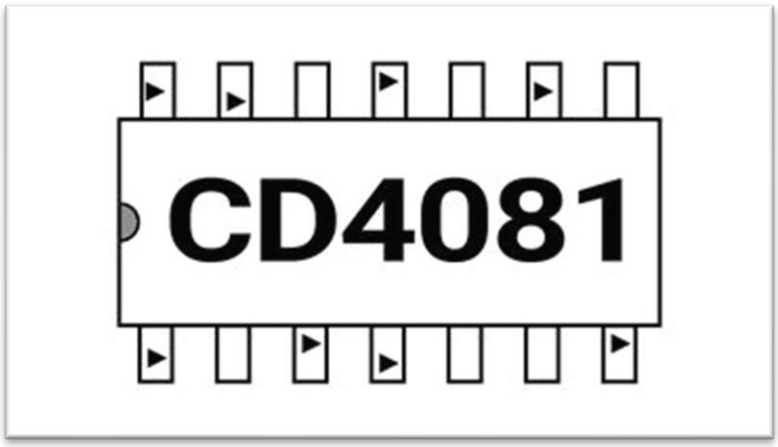
The target image for the IC 4001 topic to show the 3D model

IC 4011 NAND Gate Target Image

The target image for the IC 4011 topic to show the 3D model

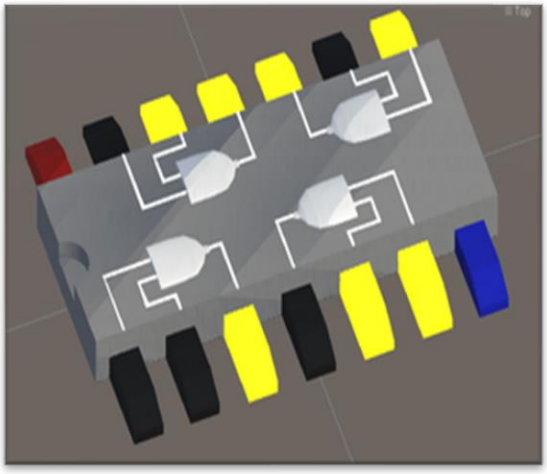
IC 4071 OR Gate Target Image

The target image for the IC 4071 topic to show the 3D model



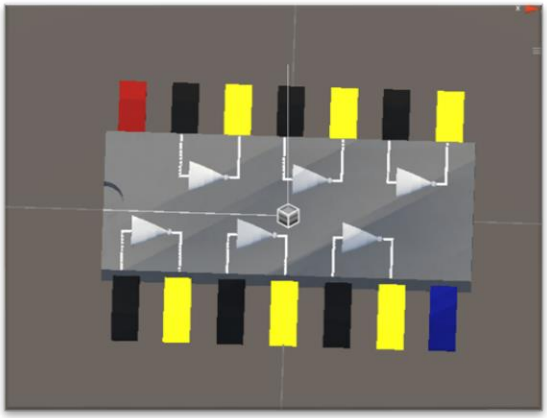
IC 4081 AND Gate Target Image

The target image for the IC 4081 topic to show the 3D model



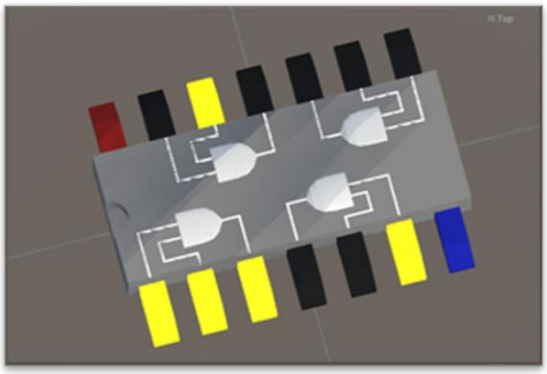
IC 4011 3D Model

The 3D model of the IC 4011 that is shown when the target image is recognized



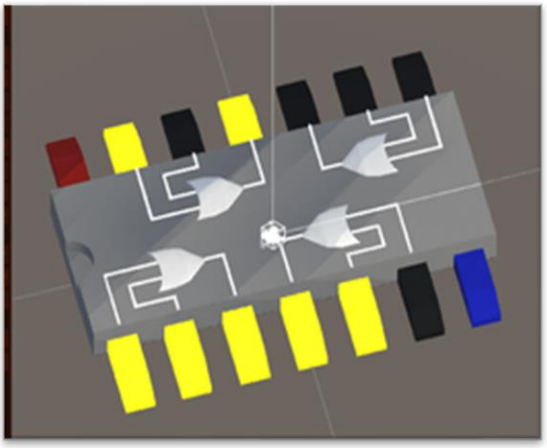
IC 4069 3D Model

The 3D model of the IC 4069 that is shown when the target image is recognized



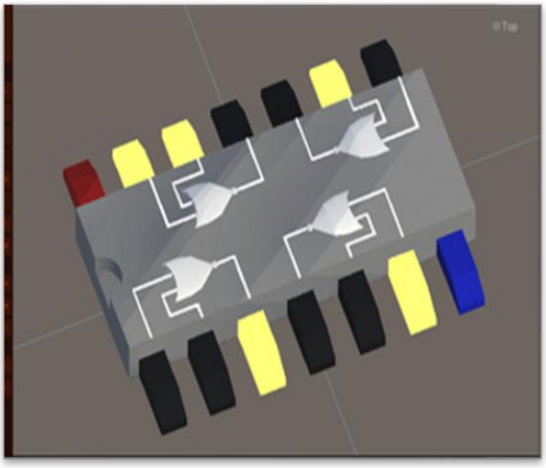
IC 4081 3D Model

The 3D model of the IC 4081 that is shown when the target image is recognized



IC 4071 3D Model

The 3D model of the IC 4071 that is shown when the target image is recognized



IC 4001 3D Model

The 3D model of the IC 4001 that is shown when the target image is recognized

Appendix B

Evaluation Sheets

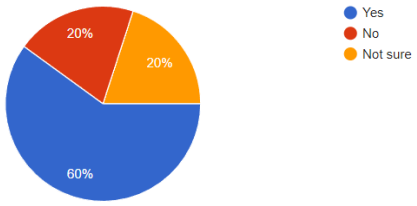
Evaluation Tools (Using Google Forms)

Demo

1.Are you familiar with AR?

Copy

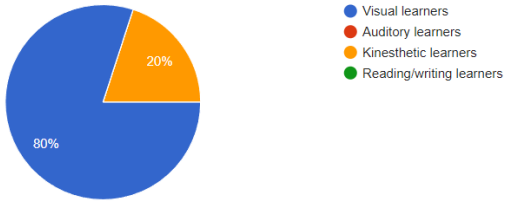
5 responses



2. Among the 4 types of Learning Styles which do you prefer?

Copy

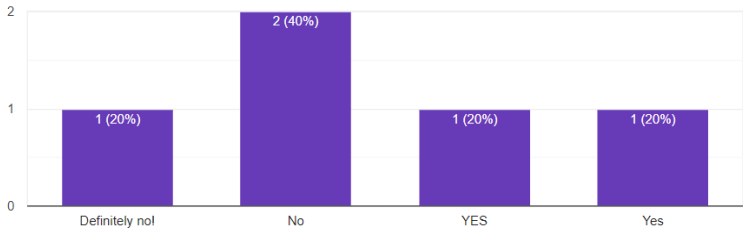
5 responses



3.You think Augmented Reality is easy to use as a user?

Copy

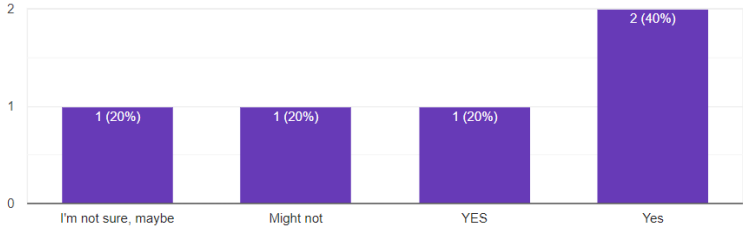
5 responses



4.When using the Augmented Reality App have you better understood the topic?

Copy

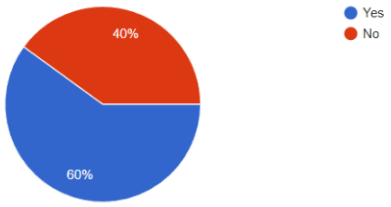
5 responses



5. Do you think the use of AR APP would have a positive effect inside the classroom?

Copy

5 responses



6.What topics would you like to see in our app?

4 responses

About computers.

Games

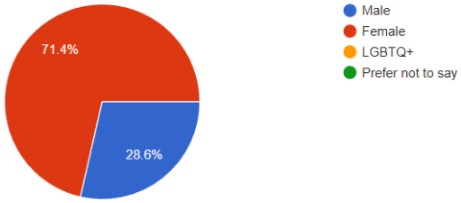
GIPHY WORLD

I think the topic that I would like to see is the parts of computer hardware.

Gender

Copy

7 responses



Course

Copy

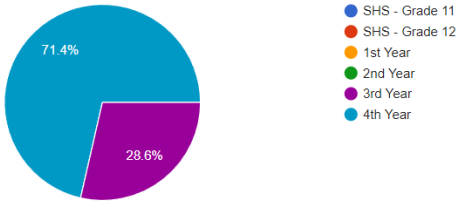
7 responses

Students

Year

7 responses

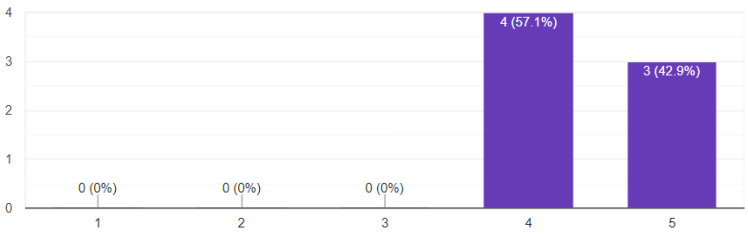
Copy



The AR ICC Visualizer App is useful in visualizing topics.

7 responses

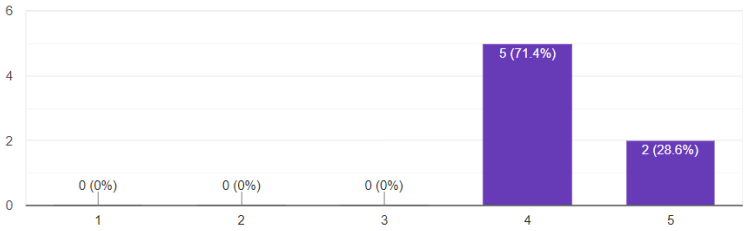
Copy



The AR ICC Visualizer App is easy to use.

7 responses

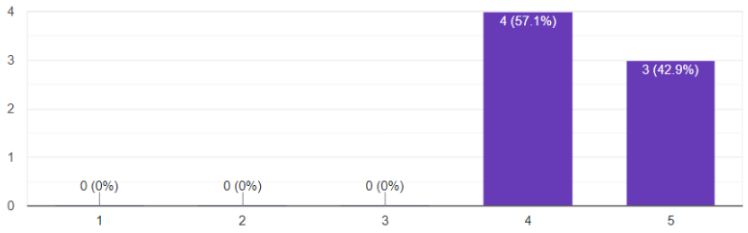
Copy

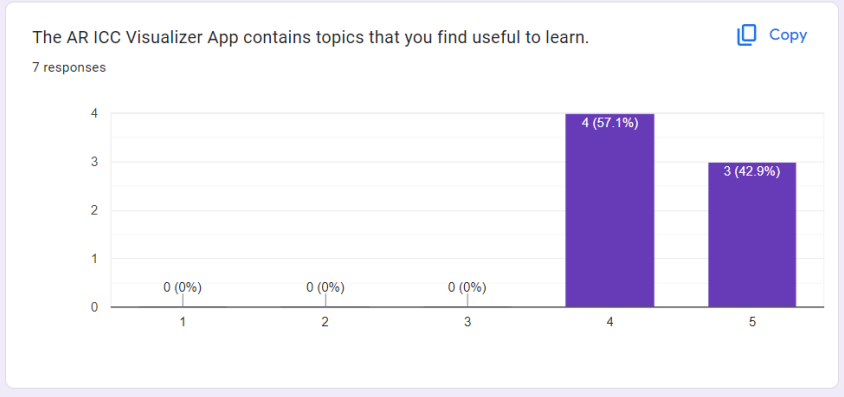


The AR ICC Visualizer App is entertaining and fun compared to the usual teaching method.

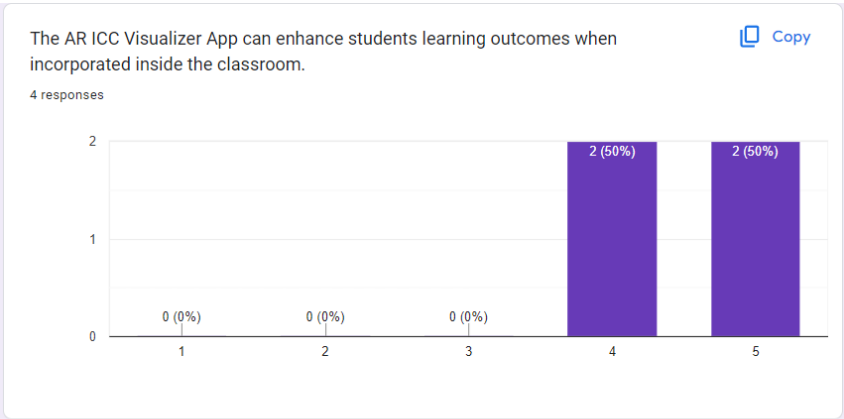
7 responses

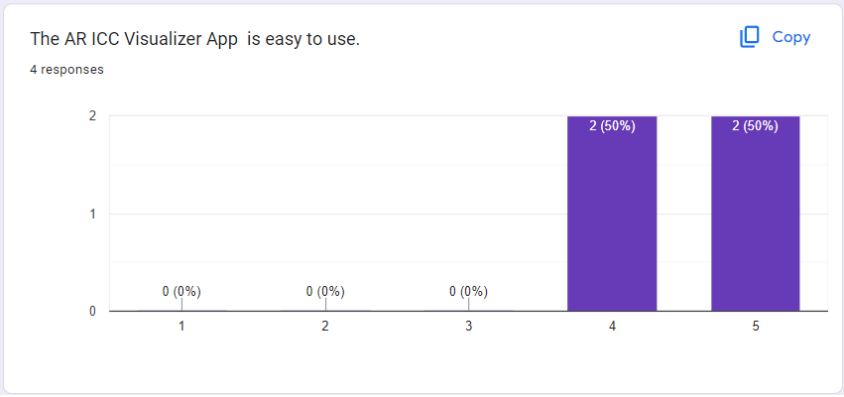
Copy




The AR ICC Visualizer App contains topics that you find useful to learn.
Copy
7 responses
4
3
2
1
0
4 (57.1%)
3 (42.9%)
0 (0%)
0 (0%)
0 (0%)
1
2
3
4
5

Teachers

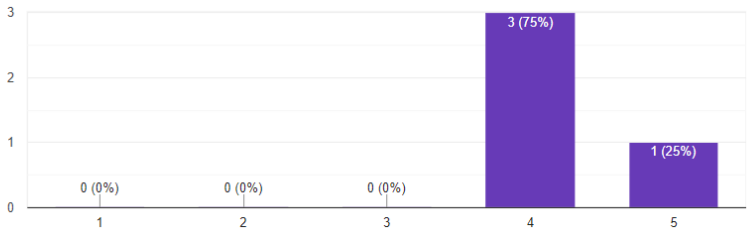

The AR ICC Visualizer App can enhance students learning outcomes when incorporated inside the classroom.
Copy
4 responses
2
1
0
2 (50%)
2 (50%)
0 (0%)
0 (0%)
0 (0%)
1
2
3
4
5


The AR ICC Visualizer App is easy to use.
Copy
4 responses
2
1
0
2 (50%)
2 (50%)
0 (0%)
0 (0%)
0 (0%)
1
2
3
4
5

The information inside the AR ICC Visualizer App is accurate enough to use as a teaching materials.

Copy

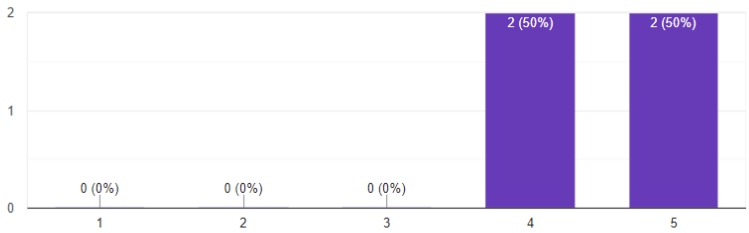
4 responses



The AR ICC Visualizer App is entertaining enough for students to prefer over the traditional teaching method.

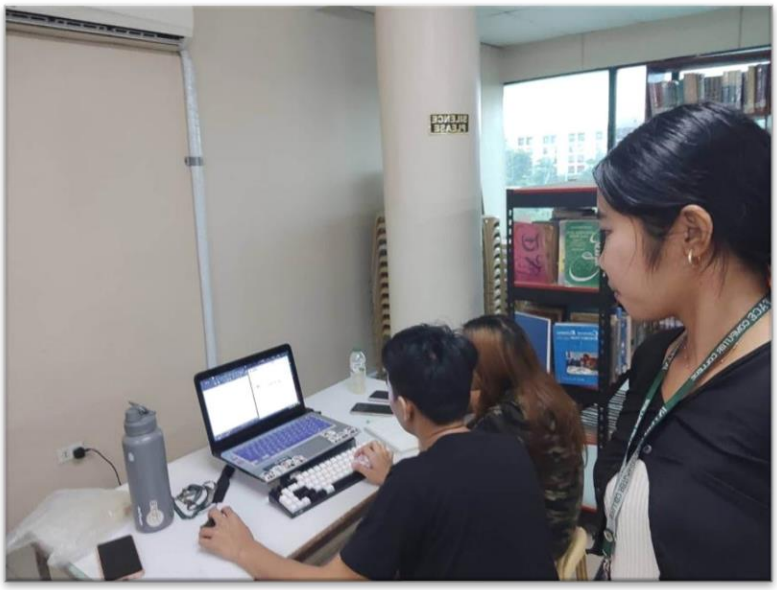
Copy

4 responses



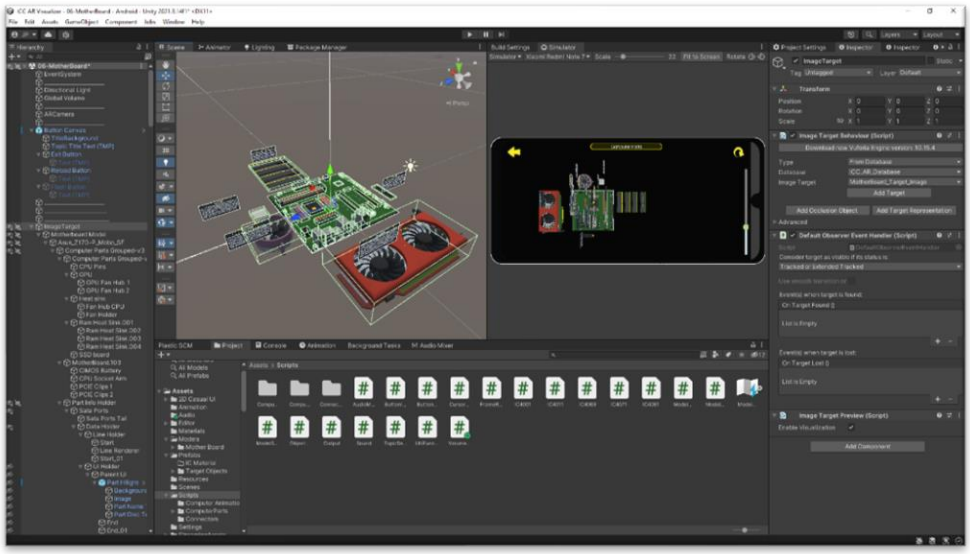
Appendix C

Supporting Documents



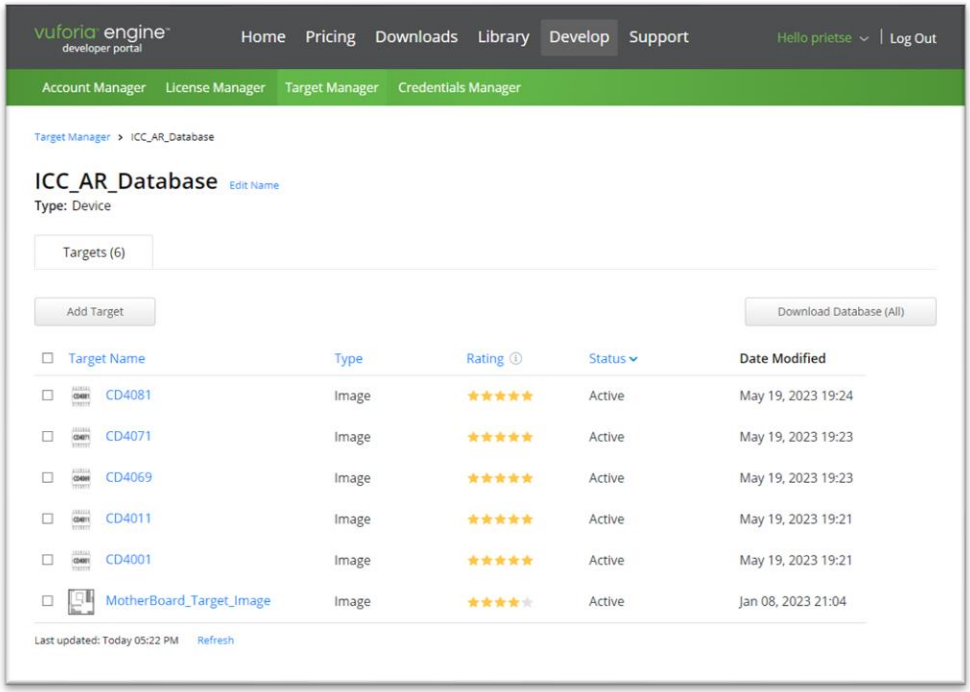
Development of the AR App

Pictures show the researchers
developing the AR application.



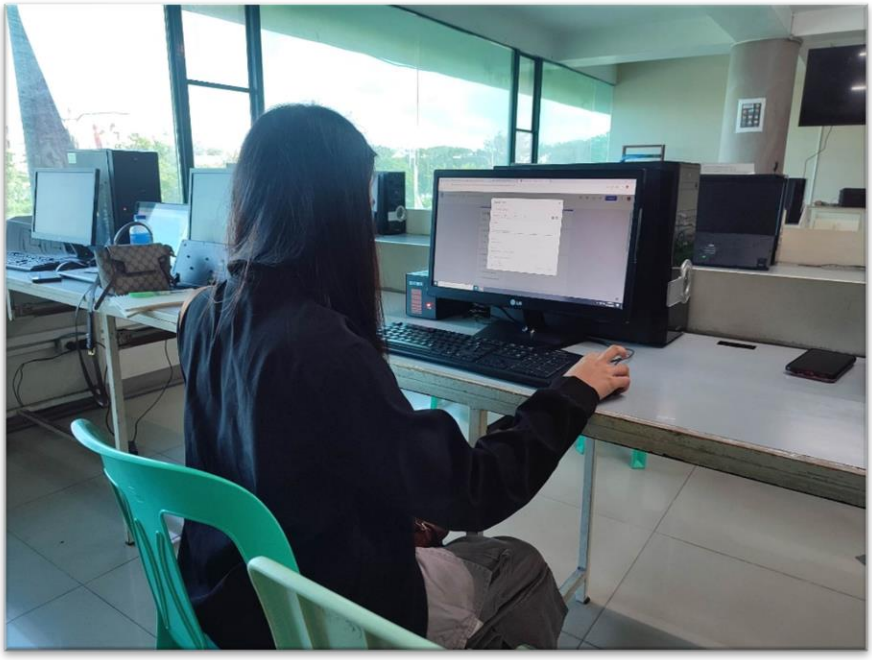
Screenshot of the Unity 3D Engine

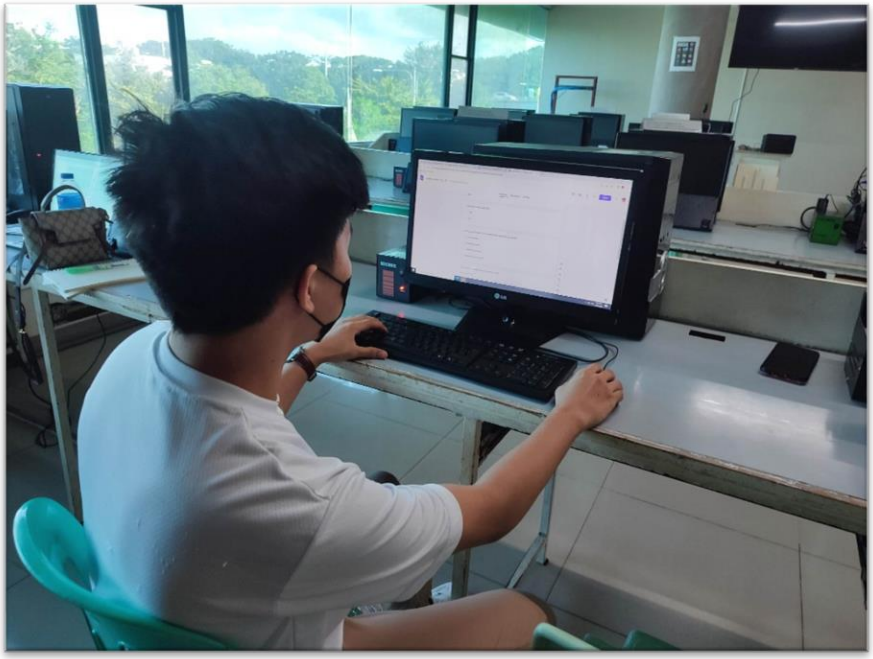
A screenshot shows the interface in which
the application was developed.



Screenshot of the Vuforia Target Manager

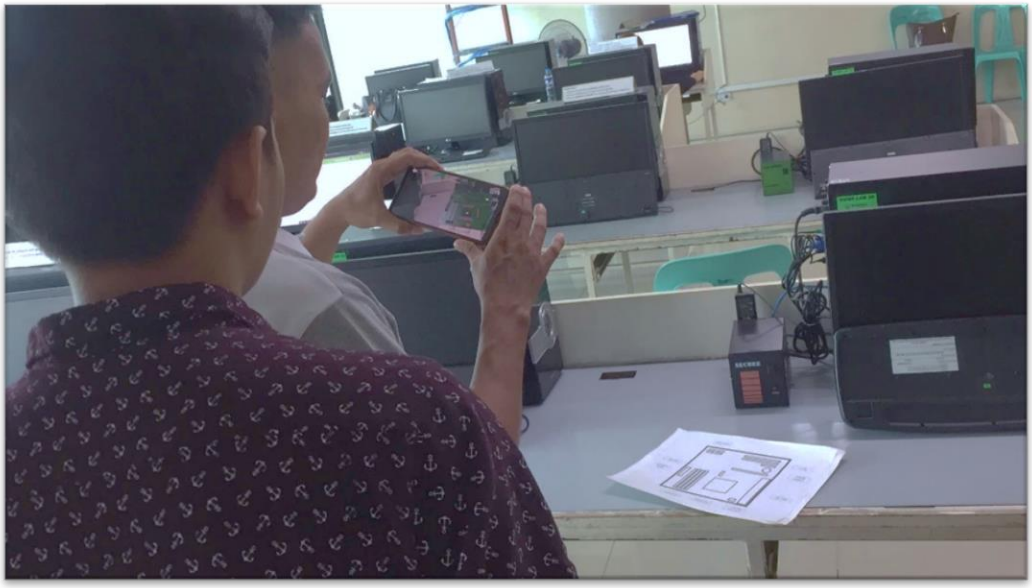
A screenshot shows the various target images used
and their compatibility with being tracked.





Survey of the AR App for ICC Students

ICC students taking the survey form after the demonstration of the AR application.



Demonstrating the AR App to a teacher of ICC

Picture of the researchers demonstration to a teacher at Interface Computer College, Inc. using the AR application and showcasing the app's capabilities.

November 25, 2022

To: Ms. Sylvia A. Villarosa
Asst. School Administrator/Principal
Interface Computer College Iloilo

Dear Ms. Villarosa:

Greetings!

In line with INTERFACE COMPUTER COLLEGE's mission of producing quality graduates, we are deploying our students to conduct an in-depth study on current systems of different establishments/institutions for their thesis proposal in **CAPSTONE 1/THESIS 1**, a requirement for the Bachelor of Science in Computer Science and Bachelor of Science in Computer Engineering (BSCS and BSCOE) program for 2022-23 School Year Semester. A group is interested in developing the **"Use of Augmented Reality in Education for Assisting Students of Interface Computer College Iloilo".**

In this connection, we are requesting your good office to allow the following students to conduct a study on the above-mentioned thesis.

Panizales, Clarence Lovely D.
Ruivivar, Lindsay
Tacaisan, Jhon Prietse D.

We will highly appreciate all the help that you can extend to our students. Rest assured that all information gathered would be kept confidential and will be intended only for the development of their thesis.

Thank you very much. God bless.

Benedic C. Mallari, MIT
Project Coordinator

Noted by:

Ms. Sylvia A. Villarosa
Administrator

Letter for Client (School)

Appendix D

User's Guide

1

Click the "Topics" button and select your desired topic.

2

Point the device towards the target image so that the target image is fully visible on the screen.

3.1

When Logic Gate topics are selected, interact with the pins by tapping the pins on the physical target image to change the input of the gates on the screen.

3.2

If you choose the "Computer Parts" topic, point towards the parts to display information about the selected part.

3.3

If the "Computer Assembly" topic is selected, point towards a part and tap the indicated button to demonstrate how it is mounted or dismounted from the motherboard.

Appendix E

Source Code

```
using UnityEngine;
namespace Computer_Animation
{
    public class AssembleAnimationManager : MonoBehaviour
    {
        [SerializeField] private Animator CPU_Animator;
        [SerializeField] private string CPU_TriggerName;
        [Space]
        [SerializeField] private Animator GPU_Animator;
        [SerializeField] private string GPU_TriggerName;
        [Space]
        [SerializeField] private Animator RAM_Animator;
        [SerializeField] private string RAM_TriggerName;
        [Space]
        [SerializeField] private Animator SSD_Animator;
        [SerializeField] private string SSD_TriggerName;

        public void CPU_Animation()
        {
            var status =
CPU_Animator.GetBool(CPU_TriggerName);
            if
(!(CPU_Animator.GetCurrentAnimatorStateInfo(0).normalizedTi
me < 1))
            {
                CPU_Animator.SetBool(CPU_TriggerName,
!status);
            }
        }
        public void GPU_Animation()
        {
            var status =
GPU_Animator.GetBool(GPU_TriggerName);
            if
(!(GPU_Animator.GetCurrentAnimatorStateInfo(0).normalizedTi
me < 1))
            {
                GPU_Animator.SetBool(GPU_TriggerName,
!status);
            }
        }
```

```
        public void RAM_Animation()
        {
            var status =
RAM_Animator.GetBool(RAM_TriggerName);
            if
(!(RAM_Animator.GetCurrentAnimatorStateInfo(0).normalizedTi
me < 1))
            {
                RAM_Animator.SetBool(RAM_TriggerName,
!status);
            }
        }


        public void SSD_Animation()
        {
            var status =
SSD_Animator.GetBool(SSD_TriggerName);
            if
(!(SSD_Animator.GetCurrentAnimatorStateInfo(0).normalizedTi
me < 1))
            {
                SSD_Animator.SetBool(SSD_TriggerName,
!status);
            }
        }
    }
}
```

```
using UnityEngine;
namespace ComputerParts
{
    public class CameraRayCaster : MonoBehaviour
    {
        [SerializeField] private float range = 10f;
        [SerializeField] private Transform ArCamera;
        [SerializeField] private GameObject cursor;
        [SerializeField] private LayerMask cursorLayer;
        [SerializeField] private bool showDebugRay;
        private RayCaster rayCaster;


        private void Start()
        {
            rayCaster = new RayCaster();
            rayCaster.OnRayEnter += CursorEntry;
            rayCaster.OnRayExit += CursorExit;
            rayCaster.OnRayStay += CursorStay;
            rayCaster.Direction = Vector3.forward;
            rayCaster.LayerMask = cursorLayer;
            rayCaster.StartTransform = ArCamera;
            rayCaster.RayLength = range;
        }


        void Update()
        {
            rayCaster.CastRay();
            rayCaster.DebugRay(showDebugRay);
        }


        private void CursorEntry(Collider _collider)
        {
            if (!_collider.CompareTag("MotherBoard"))
                return;
            if (!cursor.activeSelf)
                cursor.SetActive(true);
        }


        private void CursorStay(Collider _collider)
        {

            if (!_collider.CompareTag("MotherBoard"))
                return;
```

```
            cursor.transform.position =
rayCaster.GetHitPoint();
        }

        private void CursorExit(Collider _collider)
        {
            if (!_collider.CompareTag("MotherBoard"))
                return;
            if (cursor.activeSelf)
                cursor.SetActive(false);
        }
    }
}
```

```
using UnityEngine;
namespace ComputerParts
{
    public class LineRendererController : MonoBehaviour
    {
        private UnityEngine.LineRenderer lineRenderer;
        private Transform[] points;


        private void Awake()
        {
            lineRenderer =
GetComponent<UnityEngine.LineRenderer>();
        }
        public void SetUpLine(Transform[] _points)
        {
            lineRenderer.positionCount = _points.Length;
            this.points = _points;
        }


        void Update()
        {
            for (int i = 0; i < points.Length; i++)
            {

lineRenderer.SetPosition(i,points[i].position);
            }
        }
    }
}
```

```
using UnityEngine;
namespace ComputerParts
{
    public class LineSetter : MonoBehaviour
    {
        [SerializeField] private Transform[] points;
        [SerializeField] private LineRendererController
line;

        void Start()
        {
            line.SetUpLine(points);
        }
    }
}
```

```
using UnityEngine;
[CreateAssetMenu(fileName = "Data", menuName =
"ScriptableObjects/PartData")]
public class PartData : ScriptableObject
{
   public string PartName;
   [TextArea(15,20)]
   public string PartDescription;
}
```

```
using TMPro;
using UnityEngine;
namespace ComputerParts
{
    public class PartsHighlighter : MonoBehaviour
    {
        public bool isSelected;
        private MeshRenderer meshRenderer;
        [SerializeField] private GameObject dataHolder;
        [SerializeField] private PartData partData;
        [SerializeField] private TextMeshProUGUI partName;
        [SerializeField] private TextMeshProUGUI partDisc;


        void Start()
        {
            meshRenderer = GetComponent<MeshRenderer>();
            if (dataHolder!=null)
            {
                partName.text = partData.PartName;
                partDisc.text = partData.PartDescription;
            }
        }


        void Update()
        {
            Selected();
        }
        private void Selected()
        {
            if (isSelected)
            {
                if (dataHolder!=null)
                {
                    dataHolder.SetActive(true);
                }

                if (!meshRenderer.enabled)
                {
                    if
(meshRenderer.gameObject.transform.childCount >= 1)
                    {
                        for (int i = 0; i <
meshRenderer.transform.childCount; i++)
                        {
```

```
if
(meshRenderer.transform.GetChild(i).GetComponent<MeshRender
er>() == null)
                                    continue;
                                var child =
meshRenderer.transform.GetChild(i);

child.GetComponent<MeshRenderer>().enabled = true;
                            }
                        }
                        meshRenderer.enabled = true;
                    }
                }
                else
                {
                    if (dataHolder != null)
                    {
                        dataHolder.SetActive(false);
                    }
                    if (meshRenderer.enabled)
                    {
                        if
(meshRenderer.gameObject.transform.childCount >= 1)
                        {
                            for (int i = 0; i <
meshRenderer.transform.childCount; i++)
                            {
                                if
(meshRenderer.transform.GetChild(i).GetComponent<MeshRender
er>() == null)
                                    continue;
                                var child =
meshRenderer.transform.GetChild(i);

child.GetComponent<MeshRenderer>().enabled = false;
                            }
                        }
                        meshRenderer.enabled = false;
                    }
                }
            }
        }
}
```

```
using UnityEngine;
namespace ComputerParts
{
    public class PartsRayCaster : MonoBehaviour
    {
        [SerializeField] private float range = 10f;
        [SerializeField] private Transform ArCamera;
        [SerializeField] private LayerMask objectLayer;
        [SerializeField] private bool showDebugRay;
        private RayCaster rayCaster;
        void Start()
        {
            rayCaster = new RayCaster();
            rayCaster.OnRayEnter += RayEntry;
            rayCaster.OnRayExit += RayExit;
            rayCaster.Direction = Vector3.forward;
            rayCaster.LayerMask = objectLayer;
            rayCaster.StartTransform = ArCamera;
            rayCaster.RayLength = range;
        }
        void Update()
        {
            rayCaster.CastRay();
            rayCaster.DebugRay(showDebugRay);
        }
        private void RayEntry(Collider _collider)
        {

_collider.GetComponent<PartsHighlighter>().isSelected =
true;
        }
        private void RayExit(Collider _collider)
        {

_collider.GetComponent<PartsHighlighter>().isSelected =
false;
        }
    }
}
```

```
using System;
using UnityEngine;
namespace ComputerParts
{
    public class RayCaster {

        public Transform StartTransform;
        public Transform EndTransform;
        public Vector3 Direction;
        public float RayLength;
        public int LayerMask;

        public event Action<Collider> OnRayEnter;
        public event Action<Collider> OnRayStay;
        public event Action<Collider> OnRayExit;

        public Vector3 HitPoint;

        Collider previous;
        RaycastHit hit = new RaycastHit();

        public bool CastRay() {
            var ray = new Ray(StartTransform.position,
StartTransform.TransformDirection(Direction * RayLength));
            Physics.Raycast(ray, out hit, RayLength,
LayerMask);
            ProcessCollision(hit.collider);
            HitPoint = hit.point;
            return hit.collider != null ? true : false;
        }
        public bool CastLine() {
            Physics.Linecast(StartTransform.position,
EndTransform.position, out hit, LayerMask);
            ProcessCollision(hit.collider);
            HitPoint = hit.point;
            return hit.collider != null ? true : false;
        }


        private void ProcessCollision(Collider current) {
            if (current == null) {
                if (previous != null) {
                    DoEvent(OnRayExit, previous);
                }
            }
```

```
            else if (previous == current) {
                DoEvent(OnRayStay, current);
            }

            else if (previous != null) {
                DoEvent(OnRayExit, previous);
                DoEvent(OnRayEnter, current);
            }

            else {
                DoEvent(OnRayEnter, current);
            }

            previous = current;
        }


        private void DoEvent(Action<Collider> action,
Collider collider) {
            if (action != null) {
                action(collider);
            }
        }

        public static int GetLayerMask(LayerMask layerName,
int existingMask=0)
        {
            int layer = layerName;
            return existingMask | (1 << layer);
        }

        public Vector3 GetHitPoint()
        {
            return HitPoint;
        }

        public void DebugRay(bool _isShow)
        {
            if (_isShow)
            {
                Debug.DrawRay(StartTransform.position,
StartTransform.TransformDirection(Direction*RayLength));
            }
        }
    }
}
```

```
using UnityEngine;
namespace ComputerParts
{
    public class UIFaceCamera : MonoBehaviour
    {
        [SerializeField] private Transform ArCamera;
        [SerializeField] private Transform UI;

        void Update()
        {
            RotateCanvas();
        }

        private void RotateCanvas()
        {
            var rotation1 = ArCamera.rotation;

UI.LookAt(UI.position+rotation1*Vector3.forward,
rotation1*Vector3.up);
        }
    }
}
```

```
using System;
using UnityEngine;

public class AudioManager : MonoBehaviour
{
    public Sound[] sounds;
    public static AudioManager instance;


    void Awake()
    {
        if (instance==null)
        {
            instance = this;
        }
        else
        {
            Destroy(gameObject);
            return;
        }
        DontDestroyOnLoad(gameObject);
        foreach (var s in sounds)
        {
          s.source =
gameObject.AddComponent<AudioSource>();

          s.source.clip = s.clip;
          s.source.volume = s.volume;
          s.source.pitch = s.pitch;
          s.source.loop = s.loop;

          s.source.outputAudioMixerGroup = s.MixerGroup;
        }
    }


    private void Start()
    {
        Play("Theme");
    }

    public void Play(string clipName)
   {
        Sound s = Array.Find(sounds, sound => sound.name ==
clipName);
        if (s == null)
        {
```

```
            Debug.LogWarning("Sound: " + clipName + " not
found!");
            return;
        }
        s.source.Play();
    }

    public void Stop(string clipName)
    {
        Sound s = Array.Find(sounds, sound => sound.name ==
clipName);
        if (s == null)
        {
            Debug.LogWarning("Sound: " + clipName + " not
found!");
            return;
        }
        s.source.Stop();
    }

    public void Pause(string clipName)
    {
        Sound s = Array.Find(sounds, sound => sound.name ==
clipName);
        if (s == null)
        {
            Debug.LogWarning("Sound: " + clipName + " not
found!");
            return;
        }
        s.source.Pause();
    }

    public void UnPause(string clipName)
    {
        Sound s = Array.Find(sounds, sound => sound.name ==
clipName);
        if (s == null)
        {
            Debug.LogWarning("Sound: " + clipName + " not
found!");
            return;
        }
        s.source.UnPause();
    }
}
```

```
using UnityEngine;
using Vuforia;
public class ButtonInput : MonoBehaviour
{

    [SerializeField] private VirtualButtonBehaviour
virtualButton;
    [SerializeField] private MeshRenderer buttonMaterial;
    [SerializeField] private Color buttonColorOff;
    [SerializeField] private Color buttonColorOn;
    [SerializeField] private ConnectorOut connectorOut;
    [SerializeField] private bool signal;
    private static readonly int EmissionColor =
Shader.PropertyToID("_EmissionColor");
    private const float colorIntensity = 2f;


    private void Awake()
    {

virtualButton.RegisterOnButtonPressed(VirtualButtonPressed)
;
    }


    private void Start()
    {
        signal = false;
        ChangeButtonState(signal);
    }


    private void
VirtualButtonPressed(VirtualButtonBehaviour obj)
    {
        signal = !signal;
        ChangeButtonState(signal);
        connectorOut.Signal = signal;
    }


    private void ChangeButtonState(bool _signal)
    {
        var _vbMaterial = buttonMaterial.material;

        if (_signal)
            _vbMaterial.EnableKeyword("_EMISSION");
```

```
        else
            _vbMaterial.DisableKeyword("_EMISSION");

        _vbMaterial.SetColor(EmissionColor, _signal ?
buttonColorOn*colorIntensity :
buttonColorOff*colorIntensity);
        _vbMaterial.color = _signal ? buttonColorOn :
buttonColorOff;    }
}
```

```
using Unity.VisualScripting;
using UnityEngine;
using UnityEngine.SceneManagement;
using Vuforia;

public class ButtonManager : MonoBehaviour
{
    [Header("Audio")]
    [SerializeField] private string clickName;
    private bool isFlashOn;

    public void TurnFlashOn()
    {
        print($"{isFlashOn} flash status");

VuforiaBehaviour.Instance.CameraDevice.SetFlash(isFlashOn);
        isFlashOn = !isFlashOn;
    }

    public void LoadScene(int sceneIndex)
    {
        ClickSound();
        SceneManager.LoadScene(sceneIndex);
    }

    public void ReloadScene()
    {
        ClickSound();
        Scene s = SceneManager.GetActiveScene();
        SceneManager.LoadScene(s.buildIndex);
    }

    public void NextScene()
    {
        ClickSound();
        Scene s = SceneManager.GetActiveScene();
        SceneManager.LoadScene(s.buildIndex+1);
    }

    public void ExitGame()
    {
        ClickSound();
        Application.Quit();
    }
```

```
    public void HoverEnterButton(Object btn)
    {
        var s =
btn.GameObject().transform.localScale.z+0.05f;
        btn.GameObject().transform.localScale = new
Vector3(s,s,s);
    }

    public void HoverExitButton(Object btn)
    {
        btn.GameObject().transform.localScale = new
Vector3(1,1,1);
    }

    public void PauseTime()
    {
        Time.timeScale = 0;
    }

    public void ResumeTime()
    {
        Time.timeScale = 1;
    }

    public void ClickSound()
    {
        AudioManager.instance.Play(clickName);
    }
}
```

```
using UnityEngine;
public class CursorCheck : MonoBehaviour
{
    [SerializeField] private GameObject button_CPU;
    [SerializeField] private GameObject button_RAM;
    [SerializeField] private GameObject button_GPU;
    [SerializeField] private GameObject button_SSD;

    private void Start()
    {
        button_CPU.SetActive(false);
        button_GPU.SetActive(false);
        button_RAM.SetActive(false);
        button_SSD.SetActive(false);
    }

    private void OnTriggerEnter(Collider other)
    {
        if (other.gameObject.name.Contains("CPU"))
        {
            button_CPU.SetActive(true);
            // Debug.Log("Collided with CPU!");
        }
        else if  (other.gameObject.name.Contains("GPU"))
        {
            button_GPU.SetActive(true);
            // Debug.Log("Collided with GPU!");
        }
        else if  (other.gameObject.name.Contains("RAM"))
        {
            button_RAM.SetActive(true);
            // Debug.Log("Collided with RAM!");
        }
        else if  (other.gameObject.name.Contains("SSD"))
        {
            button_SSD.SetActive(true);
            // Debug.Log("Collided with SSD!");
        }
        else
        {
            button_CPU.SetActive(false);
            button_GPU.SetActive(false);
            button_RAM.SetActive(false);
            button_SSD.SetActive(false);
        }
    }
```

```
    private void OnTriggerExit(Collider other)
    {
        if (other.gameObject.name.Contains("CPU"))
        {
            button_CPU.SetActive(false);
            // Debug.Log("Collided with CPU!");
        }
        else if  (other.gameObject.name.Contains("GPU"))
        {
            button_GPU.SetActive(false);
            // Debug.Log("Collided with GPU!");
        }
        else if  (other.gameObject.name.Contains("RAM"))
        {
            button_RAM.SetActive(false);
            // Debug.Log("Collided with RAM!");
        }
        else if  (other.gameObject.name.Contains("SSD"))
        {
            button_SSD.SetActive(false);
            // Debug.Log("Collided with SSD!");
        }
        else
        {
            button_CPU.SetActive(false);
            button_GPU.SetActive(false);
            button_RAM.SetActive(false);
            button_SSD.SetActive(false);
            Debug.Log("Collided with a different object!");
        }
    }

    private void OnDisable()
    {
        button_CPU.SetActive(false);
        button_GPU.SetActive(false);
        button_RAM.SetActive(false);
        button_SSD.SetActive(false);
    }
}
```

```
using UnityEngine;
using Vuforia;

public class FrameRateSettings : MonoBehaviour
{
    void Start()
    {
        VuforiaApplication.Instance.OnVuforiaStarted +=
OnVuforiaStarted;
    }

    void OnVuforiaStarted()
    {
        // Query Vuforia for recommended frame rate and set
it in Unity
        var targetFps =
VuforiaBehaviour.Instance.CameraDevice.GetRecommendedFPS();

        // By default, we use Application.targetFrameRate
to set the recommended frame rate.
        // If developers use vsync in their quality
settings,
        // they should also set their
QualitySettings.vSyncCount
        // according to the value returned above.
        // e.g: If targetFPS > 50 --> vSyncCount = 1; else
vSyncCount = 2;
        if (UnityEngine.Application.targetFrameRate !=
targetFps)
        {
            Debug.Log("Setting frame rate to " + targetFps
+ "fps");
            UnityEngine.Application.targetFrameRate =
targetFps;
        }
    }
}
```

```
using UnityEngine;
using Vuforia;

public class IC4001 : MonoBehaviour
{

    [SerializeField] private VirtualButtonBehaviour
virtualButton1;
    [SerializeField] private VirtualButtonBehaviour
virtualButton2;
    [SerializeField] private MeshRenderer pinMaterialIn1;
    [SerializeField] private MeshRenderer pinMaterialIn2;
    [SerializeField] private MeshRenderer pinMaterialOut;
    [SerializeField] private Color pinColorOff;
    [SerializeField] private Color pinColorOn;
    [SerializeField] private bool signal1;
    [SerializeField] private bool signal2;
    private static readonly int EmissionColor =
Shader.PropertyToID("_EmissionColor");
    private const float colorIntensity = 2f;
    [SerializeField] private bool debug;
    private void Awake()
    {

virtualButton1.RegisterOnButtonPressed(VirtualButtonPressed
1);

virtualButton2.RegisterOnButtonPressed(VirtualButtonPressed
2);
    }
    void Start()
    {
        signal1 = false;
        signal2 = false;
        NORLogic(signal1, signal2,pinMaterialOut);
    }
    private void
VirtualButtonPressed1(VirtualButtonBehaviour obj)
    {
        AudioManager.instance.Play("click");
        signal1 = !signal1;
        ChangePinColor(signal1, pinMaterialIn1);
        NORLogic(signal1, signal2,pinMaterialOut);
    }

    private void
VirtualButtonPressed2(VirtualButtonBehaviour obj)
```

```
    {
        AudioManager.instance.Play("click");
        signal2 = !signal2;
        ChangePinColor(signal2, pinMaterialIn2);
        NORLogic(signal1, signal2,pinMaterialOut);
    }
    private void NORLogic(bool _signal1, bool _signal2,
MeshRenderer _meshRenderer)
    {
        var _outSignal = _signal1 || _signal2;
        ChangePinColor(!_outSignal , _meshRenderer);
    }
    private void ChangePinColor(bool _signal, MeshRenderer
_meshRenderer)
    {
        if (_signal)
        {
            foreach (var m in _meshRenderer.materials)
            {
                m.SetColor(EmissionColor,
pinColorOn*colorIntensity);
                m.EnableKeyword("_EMISSION");
                m.color = pinColorOn;
            }
        }
        else
        {
            foreach (var m in _meshRenderer.materials)
            {
                m.SetColor(EmissionColor,
pinColorOff*colorIntensity);
                m.DisableKeyword("_EMISSION");
                m.color = pinColorOff;
            }
        }
    }
    void Update()
    {
        if (debug)
        {
            ChangePinColor(signal1, pinMaterialIn1);
            ChangePinColor(signal2, pinMaterialIn2);
            NORLogic(signal1, signal2,pinMaterialOut);
        }
    }
}
```

```
using UnityEngine;
using Vuforia;

public class IC4011 : MonoBehaviour
{
     [SerializeField] private VirtualButtonBehaviour
virtualButton1;
    [SerializeField] private VirtualButtonBehaviour
virtualButton2;
    [SerializeField] private MeshRenderer pinMaterialIn1;
    [SerializeField] private MeshRenderer pinMaterialIn2;
    [SerializeField] private MeshRenderer pinMaterialOut;
    [SerializeField] private Color pinColorOff;
    [SerializeField] private Color pinColorOn;
    [SerializeField] private bool signal1;
    [SerializeField] private bool signal2;
    private static readonly int EmissionColor =
Shader.PropertyToID("_EmissionColor");
    private const float colorIntensity = 2f;
    [SerializeField] private bool debug;
    private void Awake()
    {

virtualButton1.RegisterOnButtonPressed(VirtualButtonPressed
1);

virtualButton2.RegisterOnButtonPressed(VirtualButtonPressed
2);
    }
    void Start()
    {
        signal1 = false;
        signal2 = false;
        NANDLogic(signal1, signal2,pinMaterialOut);
    }
    private void
VirtualButtonPressed1(VirtualButtonBehaviour obj)
    {
        AudioManager.instance.Play("click");
        signal1 = !signal1;
        ChangePinColor(signal1, pinMaterialIn1);
        NANDLogic(signal1, signal2,pinMaterialOut);
    }
    private void
VirtualButtonPressed2(VirtualButtonBehaviour obj)
    {
        AudioManager.instance.Play("click");
```

```
            signal2 = !signal2;
            ChangePinColor(signal2, pinMaterialIn2);
            NANDLogic(signal1, signal2,pinMaterialOut);
        }
        private void NANDLogic(bool _signal1, bool _signal2,
MeshRenderer _meshRenderer)
        {
            var _outSignal = _signal1 && _signal2;
            ChangePinColor(!_outSignal, _meshRenderer);
        }
        private void ChangePinColor(bool _signal, MeshRenderer
_meshRenderer)
        {
            if (_signal)
            {
                foreach (var m in _meshRenderer.materials)
                {
                    m.SetColor(EmissionColor,
pinColorOn*colorIntensity);
                    m.EnableKeyword("_EMISSION");
                    m.color = pinColorOn;
                }
            }
            else
            {
                foreach (var m in _meshRenderer.materials)
                {
                    m.SetColor(EmissionColor,
pinColorOff*colorIntensity);
                    m.DisableKeyword("_EMISSION");
                    m.color = pinColorOff;
                }
            }
        }

        void Update()
        {
            if (debug)
            {
                ChangePinColor(signal1, pinMaterialIn1);
                ChangePinColor(signal2, pinMaterialIn2);
                NANDLogic(signal1, signal2,pinMaterialOut);
            }
        }
}
```

```
using UnityEngine;
using Vuforia;

public class IC4069 : MonoBehaviour
{
    [SerializeField] private VirtualButtonBehaviour
virtualButton;
    [SerializeField] private MeshRenderer pinMaterialIn;
    [SerializeField] private MeshRenderer pinMaterialOut;
    [SerializeField] private Color pinColorOff;
    [SerializeField] private Color pinColorOn;
    [SerializeField] private bool signal;
    [SerializeField] private bool debug;
    private static readonly int EmissionColor =
Shader.PropertyToID("_EmissionColor");
    private const float colorIntensity = 2f;
    private void Awake()
    {

virtualButton.RegisterOnButtonPressed(VirtualButtonPressed)
;
    }
    private void Start()
    {
        signal = false;
        ChangePinColor(signal,pinMaterialIn);
        NotLogic(signal, pinMaterialOut);
    }
    private void
VirtualButtonPressed(VirtualButtonBehaviour obj)
    {
        AudioManager.instance.Play("click");
        signal = !signal;
        ChangePinColor(signal, pinMaterialIn);
        NotLogic(signal, pinMaterialOut);
    }
    private void NotLogic(bool _signal, MeshRenderer
_meshRenderer)
    {
        _signal = !_signal;
        ChangePinColor(_signal, _meshRenderer);
    }
    private void ChangePinColor(bool _signal, MeshRenderer
_meshRenderer)
    {
```

```
        if (_signal)
        {
            foreach (var m in _meshRenderer.materials)
            {
                m.SetColor(EmissionColor,
pinColorOn*colorIntensity);
                m.EnableKeyword("_EMISSION");
                m.color = pinColorOn;
            }
        }
        else
        {
            foreach (var m in _meshRenderer.materials)
            {
                m.SetColor(EmissionColor,
pinColorOff*colorIntensity);
                m.DisableKeyword("_EMISSION");
                m.color = pinColorOff;
            }
        }
    }

    private void Update()
    {
        if (debug)
        {
            ChangePinColor(signal, pinMaterialIn);
            NotLogic(signal, pinMaterialOut);
        }
    }
}
```

```
using UnityEngine;
using Vuforia;

public class IC4071 : MonoBehaviour
{
    [SerializeField] private VirtualButtonBehaviour
virtualButton1;
    [SerializeField] private VirtualButtonBehaviour
virtualButton2;
    [SerializeField] private MeshRenderer pinMaterialIn1;
    [SerializeField] private MeshRenderer pinMaterialIn2;
    [SerializeField] private MeshRenderer pinMaterialOut;
    [SerializeField] private Color pinColorOff;
    [SerializeField] private Color pinColorOn;
    [SerializeField] private bool signal1;
    [SerializeField] private bool signal2;
    private static readonly int EmissionColor =
Shader.PropertyToID("_EmissionColor");
    private const float colorIntensity = 2f;
    [SerializeField] private bool debug;

    private void Awake()
    {

virtualButton1.RegisterOnButtonPressed(VirtualButtonPressed
1);

virtualButton2.RegisterOnButtonPressed(VirtualButtonPressed
2);
    }

    void Start()
    {
        signal1 = false;
        signal2 = false;
        ORLogic(signal1, signal2,pinMaterialOut);
    }
    private void
VirtualButtonPressed1(VirtualButtonBehaviour obj)
    {
        AudioManager.instance.Play("click");
        signal1 = !signal1;
        ChangePinColor(signal1, pinMaterialIn1);
        ORLogic(signal1, signal2,pinMaterialOut);
    }
    private void
VirtualButtonPressed2(VirtualButtonBehaviour obj)
```

```
    {
        AudioManager.instance.Play("click");
        signal2 = !signal2;
        ChangePinColor(signal2, pinMaterialIn2);
        ORLogic(signal1, signal2,pinMaterialOut);
    }
    private void ORLogic(bool _signal1, bool _signal2,
MeshRenderer _meshRenderer)
    {
        ChangePinColor(_signal1||_signal2, _meshRenderer);
    }
    private void ChangePinColor(bool _signal, MeshRenderer
_meshRenderer)
    {
        if (_signal)
        {
            foreach (var m in _meshRenderer.materials)
            {
                m.SetColor(EmissionColor,
pinColorOn*colorIntensity);
                m.EnableKeyword("_EMISSION");
                m.color = pinColorOn;
            }
        }
        else
        {
            foreach (var m in _meshRenderer.materials)
            {
                m.SetColor(EmissionColor,
pinColorOff*colorIntensity);
                m.DisableKeyword("_EMISSION");
                m.color = pinColorOff;
            }
        }
    }
    void Update()
    {
        if (debug)
        {
            ChangePinColor(signal1, pinMaterialIn1);
            ChangePinColor(signal2, pinMaterialIn2);
            ORLogic(signal1, signal2,pinMaterialOut);
        }
    }
}
```

```
using UnityEngine;
using Vuforia;

public class IC4081 : MonoBehaviour
{
    [SerializeField] private VirtualButtonBehaviour
virtualButton1;
    [SerializeField] private VirtualButtonBehaviour
virtualButton2;
    [SerializeField] private MeshRenderer pinMaterialIn1;
    [SerializeField] private MeshRenderer pinMaterialIn2;
    [SerializeField] private MeshRenderer pinMaterialOut;
    [SerializeField] private Color pinColorOff;
    [SerializeField] private Color pinColorOn;
    [SerializeField] private bool signal1;
    [SerializeField] private bool signal2;
    private static readonly int EmissionColor =
Shader.PropertyToID("_EmissionColor");
    private const float colorIntensity = 2f;
    [SerializeField] private bool debug;
    private void Awake()
    {

virtualButton1.RegisterOnButtonPressed(VirtualButtonPressed
1);

virtualButton2.RegisterOnButtonPressed(VirtualButtonPressed
2);
    }
    void Start()
    {
        signal1 = false;
        signal2 = false;
        ANDLogic(signal1, signal2,pinMaterialOut);
    }
    private void
VirtualButtonPressed1(VirtualButtonBehaviour obj)
    {
        AudioManager.instance.Play("click");
        signal1 = !signal1;
        ChangePinColor(signal1, pinMaterialIn1);
        ANDLogic(signal1, signal2,pinMaterialOut);
    }
    private void
VirtualButtonPressed2(VirtualButtonBehaviour obj)
    {
        AudioManager.instance.Play("click");
```

```
        signal2 = !signal2;
        ChangePinColor(signal2, pinMaterialIn2);
        ANDLogic(signal1, signal2,pinMaterialOut);
    }
    private void ANDLogic(bool _signal1, bool _signal2,
MeshRenderer _meshRenderer)
    {
        ChangePinColor(_signal1&&_signal2, _meshRenderer);
    }
    private void ChangePinColor(bool _signal, MeshRenderer
_meshRenderer)
    {
        if (_signal)
        {
            foreach (var m in _meshRenderer.materials)
            {
                m.SetColor(EmissionColor,
pinColorOn*colorIntensity);
                m.EnableKeyword("_EMISSION");
                m.color = pinColorOn;
            }
        }
        else
        {
            foreach (var m in _meshRenderer.materials)
            {
                m.SetColor(EmissionColor,
pinColorOff*colorIntensity);
                m.DisableKeyword("_EMISSION");
                m.color = pinColorOff;
            }
        }
    }
    void Update()
    {
        if (debug)
        {
            ChangePinColor(signal1, pinMaterialIn1);
            ChangePinColor(signal2, pinMaterialIn2);
            ANDLogic(signal1, signal2,pinMaterialOut);
        }
    }
}
```

```
using UnityEngine;

public class ModelMover : MonoBehaviour
{
    [SerializeField] private Transform ModelToMove;
    [SerializeField] private float moveSpeed;
    private ModelMoverInput moverInput;
    private float addX;
    private float addY;
    void Start()
    {
        moverInput = new ModelMoverInput();
        moverInput.Rotation.Enable();
    }

    void FixedUpdate()
    {
        Vector2 inputVector2 =
moverInput.Rotation.rot.ReadValue<Vector2>();
        if (inputVector2 == Vector2.zero)
            return;
        var rotation = ModelToMove.rotation;
        if (inputVector2.x != 0)
        {
            if (inputVector2.x<0)
            {
                if (addX <= 360)
                    addX -= moveSpeed;
                else
                    addX = 0;
            }
            else
            {
                if (addX <= 360)
                    addX += moveSpeed;
                else
                    addX = 0;
            }
        }
        if (inputVector2.y != 0)
        {
            if (inputVector2.y <0)
            {
                if (addY <= 360)
                    addY -= moveSpeed;
                else
                    addY = 0;
```

```
            }
            else
            {
                if (addY <= 360)
                    addY += moveSpeed;
                else
                    addY = 0;
            }

        }
        ModelToMove.eulerAngles = new Vector3(rotation.x +
addX, rotation.y + addY, 0);
    }

    private void OnDisable()
    {
        moverInput.Rotation.Disable();
    }
}
```

```
using UnityEngine;
using UnityEngine.UI;

public class ModelScaler : MonoBehaviour
{
    [SerializeField] private Transform ModelToScale;
    [SerializeField] private Slider slider;
    [SerializeField] private float sliderMaxValue = 3f;
    [SerializeField] private float sliderMinValue = 0.5f;
    [SerializeField] private float sliderDefaultValue = 1f;
    void Start()
    {
        slider.maxValue = sliderMaxValue;
        slider.minValue = sliderMinValue;
        slider.value = sliderDefaultValue;
    }

    void Update()
    {
        float val = slider.value;
        ModelToScale.localScale = new Vector3(val, val,
val);
    }
}
```

```
using UnityEngine;

public class ObjectSpawner : MonoBehaviour
{
  [SerializeField] private GameObject spawnedObject;
  public void  SpawnObject()
  {
    Instantiate(spawnedObject,
transform.position,Quaternion.identity);
  }
}
```

```
using UnityEngine;

public class Output : MonoBehaviour
{
    [SerializeField] private MeshRenderer lightModel;
    [SerializeField] private float colorIntensity = 2f;
    [SerializeField] private Color lightOff;
    [SerializeField] private Color lightOn;
    private Material lightMaterial;
    [SerializeField] private ConnectorIn connectorIn;
    private static readonly int EmissionColor =
Shader.PropertyToID("_EmissionColor");

    void Start()
    {
        lightMaterial = lightModel.material;
    }

    void Update()
    {
        if (connectorIn.connectedToOut)
        {
            lightMaterial.EnableKeyword("_EMISSION");
            lightMaterial.SetColor(EmissionColor,
connectorIn.Signal ? lightOn * colorIntensity : lightOff *
colorIntensity);
            lightMaterial.color = connectorIn.Signal ?
lightOn : lightOff;
        }
        else
        {
            lightMaterial.color = lightOff;
            lightMaterial.DisableKeyword("_EMISSION");
        }
    }
}
```

```
using UnityEngine;
using UnityEngine.Audio;

[System.Serializable]
public class Sound
{
    public string name;
    public AudioClip clip;
    [Range(0f,3f)]
    public float volume = 1 ;
    [Range(.1f,3f)]
    public float pitch = 1;

    public bool loop;
    [HideInInspector]public AudioSource source;
    public AudioMixerGroup MixerGroup;
}
```

```
using UnityEngine;

public class TopicSelectionOrientation : MonoBehaviour
{
    [SerializeField] private GameObject landscape;
    [SerializeField] private GameObject portrait;

    void Update()
    {
        if (Input.deviceOrientation ==
DeviceOrientation.Portrait)
        {
            portrait.SetActive(true);
            landscape.SetActive(false);
        }
        else
        {
            portrait.SetActive(false);
            landscape.SetActive(true);
        }
    }
}
```

```
using Unity.Mathematics;
using UnityEngine;

public class UtilFunctions : MonoBehaviour
{

   public void RigidbodyGravity(Rigidbody obj, bool
gravity)
   {
      obj.useGravity = gravity;
   }
   public void SmoothDamp(Transform obj, Transform target,
Vector3 velocity, float speed)
   {
      obj.position = Vector3.SmoothDamp(
         obj.position,
         target.position,
         ref velocity,
         speed * Time.deltaTime);
   }

   public void Lerp(Transform obj, Transform target, float
speed)
   {
      obj.position = Vector3.Lerp(
         obj.position,
         target.position,
         speed * Time.deltaTime);
   }

   public void MoveTowards(Transform obj, Transform target,
float speed)
   {
      obj.position = Vector3.MoveTowards(
         obj.position,
         target.position,
         speed * Time.deltaTime);
   }

   public void SlerpSpin(Transform obj, float speed)
   {
      obj.rotation = Quaternion.Slerp(Quaternion.identity,
quaternion.Euler(0,180,0), speed*Time.deltaTime);
   }


   public float FindDistance(Transform obj, Transform
```

```
target)
    {
        float dist = Vector3.Distance(obj.position,
target.position);
        return dist;
    }
   public void RotateTowards(Transform obj, Transform
target, float speed)
    {
        Vector3 targetDirection = target.position -
obj.position;
        float singleStep = speed * Time.deltaTime;
        Vector3 newDirection =
Vector3.RotateTowards(transform.forward, targetDirection,
singleStep, 0.0f);
        transform.rotation =
Quaternion.LookRotation(newDirection);
    }

}
```

```
using UnityEngine;
using UnityEngine.Audio;
using UnityEngine.UI;
public class VolumeSettings : MonoBehaviour
{
    [SerializeField] private AudioMixer musicAudioMixer;
    [SerializeField] private AudioMixer sfxAudioMixer;
    [SerializeField] private Slider musicSlider;
    [SerializeField] private Slider sfxSlider;
    private float parMusic;
    private float parSFX;

    private void Start()
    {
        if (PlayerPrefs.HasKey("ParMusic"))
        {
            musicSlider.value =
PlayerPrefs.GetFloat("ParMusic");

musicAudioMixer.SetFloat("ParMusic",PlayerPrefs.GetFloat("P
arMusic"));
        }
        if (PlayerPrefs.HasKey("ParSFX"))
        {
            sfxSlider.value =
PlayerPrefs.GetFloat("ParSFX");;

sfxAudioMixer.SetFloat("ParSFX",PlayerPrefs.GetFloat("ParSF
X"));
        }
    }
    public void SetMusic(float vol)
    {
        musicAudioMixer.SetFloat("ParMusic",vol);
        parMusic = vol;
    }
    public void SetSFX(float vol)
    {
        sfxAudioMixer.SetFloat("ParSFX",vol);
        parSFX = vol;
    }
    public void SaveVolume()
    {
        PlayerPrefs.SetFloat("ParMusic", parMusic);
        PlayerPrefs.SetFloat("ParSFX", parSFX);
    }
}
```

```
%YAML 1.1
%TAG !u! tag:unity3d.com,2011:
--- !u!114 &11400000
MonoBehaviour:
  m_ObjectHideFlags: 0
  m_CorrespondingSourceObject: {fileID: 0}
  m_PrefabInstance: {fileID: 0}
  m_PrefabAsset: {fileID: 0}
  m_GameObject: {fileID: 0}
  m_Enabled: 1
  m_EditorHideFlags: 0
  m_Script: {fileID: 11500000, guid:
1b9111171376bdd46851cb2895d4e505, type: 3}
  m_Name: 8 pin
  m_EditorClassIdentifier:
  PartName: 8 Pin Connector
  PartDescription: An 8-pin connector on a motherboard is
used to provide additional
    power to the CPU on high-performance motherboards. It
is necessary for high-end
    CPUs and overclocking, as it can supply up to 150 watts
of power, which is more
    than the standard 4-pin CPU power connector can
provide.
```

```
%YAML 1.1
%TAG !u! tag:unity3d.com,2011:
--- !u!114 &11400000
MonoBehaviour:
  m_ObjectHideFlags: 0
  m_CorrespondingSourceObject: {fileID: 0}
  m_PrefabInstance: {fileID: 0}
  m_PrefabAsset: {fileID: 0}
  m_GameObject: {fileID: 0}
  m_Enabled: 1
  m_EditorHideFlags: 0
  m_Script: {fileID: 11500000, guid:
1b9111171376bdd46851cb2895d4e505, type: 3}
```

```
  m_Name: 24 pin
  m_EditorClassIdentifier:
  PartName: 24 Pin Connector
  PartDescription: A 24-pin connector on a motherboard is a
power connector that
    connects the motherboard to the power supply unit. It
supplies the motherboard
    with the voltages required for the system to operate,
such as +12V, +5V, and
    +3.3V.
```

```
%YAML 1.1
%TAG !u! tag:unity3d.com,2011:
--- !u!114 &11400000
MonoBehaviour:
  m_ObjectHideFlags: 0
  m_CorrespondingSourceObject: {fileID: 0}
  m_PrefabInstance: {fileID: 0}
  m_PrefabAsset: {fileID: 0}
  m_GameObject: {fileID: 0}
  m_Enabled: 1
  m_EditorHideFlags: 0
  m_Script: {fileID: 11500000, guid:
1b9111171376bdd46851cb2895d4e505, type: 3}
  m_Name: Chipset
  m_EditorClassIdentifier:
  PartName: B450 Chipset
  PartDescription: A chipset on a motherboard is a group of
electronic components
    that manages the flow of data between various
components of the computer, such
    as the processor, memory, and storage devices. It acts
as a communication bridge
    and determines the features and capabilities of the
motherboard.
```

```
%YAML 1.1
%TAG !u! tag:unity3d.com,2011:
--- !u!114 &11400000
MonoBehaviour:
  m_ObjectHideFlags: 0
  m_CorrespondingSourceObject: {fileID: 0}
  m_PrefabInstance: {fileID: 0}
  m_PrefabAsset: {fileID: 0}
  m_GameObject: {fileID: 0}
  m_Enabled: 1
  m_EditorHideFlags: 0
  m_Script: {fileID: 11500000, guid:
1b9111171376bdd46851cb2895d4e505, type: 3}
  m_Name: CIMOS Battery
  m_EditorClassIdentifier:
  PartName: CIMOS Battery
  PartDescription: A CMOS battery on a motherboard is a
small battery that powers
    the real-time clock (RTC) and stores the BIOS settings
of a computer. It allows
    the motherboard to retain the BIOS settings and system
clock even when the computer
    is turned off or unplugged.
```

```
%YAML 1.1
%TAG !u! tag:unity3d.com,2011:
--- !u!114 &11400000
MonoBehaviour:
  m_ObjectHideFlags: 0
  m_CorrespondingSourceObject: {fileID: 0}
  m_PrefabInstance: {fileID: 0}
  m_PrefabAsset: {fileID: 0}
  m_GameObject: {fileID: 0}
  m_Enabled: 1
  m_EditorHideFlags: 0
  m_Script: {fileID: 11500000, guid:
1b9111171376bdd46851cb2895d4e505, type: 3}
  m_Name: CPU
```

```
  m_EditorClassIdentifier:
  PartName: ' Central Processing Unit'
  PartDescription: "\rCPU stands for Central Processing
Unit, which is the primary
    component of a computer that carries out instructions
and performs calculations.
    It is also known as the brain of the computer and is
responsible for executing
    tasks related to processing, storage, and input/output
operations."
```

```
%YAML 1.1
%TAG !u! tag:unity3d.com,2011:
--- !u!114 &11400000
MonoBehaviour:
  m_ObjectHideFlags: 0
  m_CorrespondingSourceObject: {fileID: 0}
  m_PrefabInstance: {fileID: 0}
  m_PrefabAsset: {fileID: 0}
  m_GameObject: {fileID: 0}
  m_Enabled: 1
  m_EditorHideFlags: 0
  m_Script: {fileID: 11500000, guid:
1b9111171376bdd46851cb2895d4e505, type: 3}
  m_Name: CPU Heatsink Holder
  m_EditorClassIdentifier:
  PartName: CPU Cooling Bracket
  PartDescription: A CPU cooling bracket is a hardware
component that is used to
    mount a CPU cooler onto a computer's motherboard. It is
designed to hold the
    CPU cooler in place and ensure that it makes proper
contact with the CPU.
```

```
%YAML 1.1
%TAG !u! tag:unity3d.com,2011:
--- !u!114 &11400000
MonoBehaviour:
  m_ObjectHideFlags: 0
  m_CorrespondingSourceObject: {fileID: 0}
  m_PrefabInstance: {fileID: 0}
  m_PrefabAsset: {fileID: 0}
  m_GameObject: {fileID: 0}
  m_Enabled: 1
  m_EditorHideFlags: 0
  m_Script: {fileID: 11500000, guid:
1b9111171376bdd46851cb2895d4e505, type: 3}
  m_Name: CPU Socket
  m_EditorClassIdentifier:
  PartName: AM4 CPU Socket
  PartDescription: An AM4 socket on a motherboard is
designed to house an AMD Ryzen
    processor and is the current socket standard for AMD
Ryzen processors. It provides
    the physical and electrical connections between the
processor and the motherboard.
```


```
%YAML 1.1
%TAG !u! tag:unity3d.com,2011:
--- !u!114 &11400000
MonoBehaviour:
  m_ObjectHideFlags: 0
  m_CorrespondingSourceObject: {fileID: 0}
  m_PrefabInstance: {fileID: 0}
  m_PrefabAsset: {fileID: 0}
  m_GameObject: {fileID: 0}
  m_Enabled: 1
  m_EditorHideFlags: 0
  m_Script: {fileID: 11500000, guid:
1b9111171376bdd46851cb2895d4e505, type: 3}
```

```
  m_Name: Fan Header CPU
  m_EditorClassIdentifier:
  PartName: 'CPU Fan Header '
  PartDescription: A CPU fan header is a motherboard
connector that provides power
    and control to the CPU fan. The CPU fan header is
typically located near the
    CPU socket on the motherboard and is used to connect
the CPU fan to the motherboard.
```

```
%YAML 1.1
%TAG !u! tag:unity3d.com,2011:
--- !u!114 &11400000
MonoBehaviour:
  m_ObjectHideFlags: 0
  m_CorrespondingSourceObject: {fileID: 0}
  m_PrefabInstance: {fileID: 0}
  m_PrefabAsset: {fileID: 0}
  m_GameObject: {fileID: 0}
  m_Enabled: 1
  m_EditorHideFlags: 0
  m_Script: {fileID: 11500000, guid:
1b9111171376bdd46851cb2895d4e505, type: 3}
  m_Name: Fan Header System
  m_EditorClassIdentifier:
  PartName: System Fan Header
  PartDescription: "A system fan header is a motherboard
connector that provides
    power and control to a chassis fan or system fan. The
system fan header is typically
    located near the edge of the motherboard and is used to
connect the chassis fan
    to the motherboard.\r\r"
```

```
%YAML 1.1
%TAG !u! tag:unity3d.com,2011:
--- !u!114 &11400000
MonoBehaviour:
  m_ObjectHideFlags: 0
  m_CorrespondingSourceObject: {fileID: 0}
  m_PrefabInstance: {fileID: 0}
  m_PrefabAsset: {fileID: 0}
  m_GameObject: {fileID: 0}
  m_Enabled: 1
  m_EditorHideFlags: 0
  m_Script: {fileID: 11500000, guid:
1b9111171376bdd46851cb2895d4e505, type: 3}
  m_Name: GPU
  m_EditorClassIdentifier:
  PartName: Graphics Card
  PartDescription: A graphics card, or GPU card, is a
hardware component that provides
    high-quality graphics processing. It contains a
graphics processing unit (GPU)
    and memory to render and display images, videos, and
animations. Graphics cards
    are used for gaming, video editing, and other graphics-
intensive tasks and can
    be connected to a display using interfaces such as HDMI
or DisplayPort.
```


```
%YAML 1.1
%TAG !u! tag:unity3d.com,2011:
--- !u!114 &11400000
MonoBehaviour:
  m_ObjectHideFlags: 0
  m_CorrespondingSourceObject: {fileID: 0}
  m_PrefabInstance: {fileID: 0}
  m_PrefabAsset: {fileID: 0}
  m_GameObject: {fileID: 0}
  m_Enabled: 1
  m_EditorHideFlags: 0
```

```
  m_Script: {fileID: 11500000, guid:
1b9111171376bdd46851cb2895d4e505, type: 3}
  m_Name: Heatsink
  m_EditorClassIdentifier:
  PartName: CPU Heatsink
  PartDescription: A CPU heatsink is a vital component that
sits atop the processor,
    absorbing and dissipating heat. It consists of metal
fins and is typically paired
    with a fan to enhance cooling. This setup, known as a
CPU cooler, ensures optimal
    performance and longevity for the processor.
```

```
%YAML 1.1
%TAG !u! tag:unity3d.com,2011:
--- !u!114 &11400000
MonoBehaviour:
  m_ObjectHideFlags: 0
  m_CorrespondingSourceObject: {fileID: 0}
  m_PrefabInstance: {fileID: 0}
  m_PrefabAsset: {fileID: 0}
  m_GameObject: {fileID: 0}
  m_Enabled: 1
  m_EditorHideFlags: 0
  m_Script: {fileID: 11500000, guid:
1b9111171376bdd46851cb2895d4e505, type: 3}
  m_Name: IO Audio
  m_EditorClassIdentifier:
  PartName: Audio Ports
  PartDescription: An audio panel port is a set of jacks on
a motherboard used to
    connect external audio devices. It includes jacks for
line in, line out, microphone,
    and S/PDIF, and the audio quality can vary depending on
the quality of the audio
    codec and connected devices.
```

```
%YAML 1.1
%TAG !u! tag:unity3d.com,2011:
--- !u!114 &11400000
MonoBehaviour:
  m_ObjectHideFlags: 0
  m_CorrespondingSourceObject: {fileID: 0}
  m_PrefabInstance: {fileID: 0}
  m_PrefabAsset: {fileID: 0}
  m_GameObject: {fileID: 0}
  m_Enabled: 1
  m_EditorHideFlags: 0
  m_Script: {fileID: 11500000, guid:
1b9111171376bdd46851cb2895d4e505, type: 3}
  m_Name: IO Bios
  m_EditorClassIdentifier:
  PartName: Flash BIOS
  PartDescription: 'BIOS Flashback allows updating BIOS
firmware on some motherboards
    without a working CPU, memory, or graphics card. Users
download firmware to a
    USB drive, insert it into a dedicated port, and
initiate the update. '
```

```
%YAML 1.1
%TAG !u! tag:unity3d.com,2011:
--- !u!114 &11400000
MonoBehaviour:
  m_ObjectHideFlags: 0
  m_CorrespondingSourceObject: {fileID: 0}
  m_PrefabInstance: {fileID: 0}
  m_PrefabAsset: {fileID: 0}
  m_GameObject: {fileID: 0}
  m_Enabled: 1
  m_EditorHideFlags: 0
  m_Script: {fileID: 11500000, guid:
1b9111171376bdd46851cb2895d4e505, type: 3}
```

```
  m_Name: IO Display
  m_EditorClassIdentifier:
  PartName: Display Ports
  PartDescription: A display port on a motherboard is used
to connect a computer
    display to the motherboard, allowing users to view
images, videos, and other
    graphical content. The type and number of display ports
available on a motherboard
    can vary depending on the motherboard's specifications
and capabilities.
```

```
%YAML 1.1
%TAG !u! tag:unity3d.com,2011:
--- !u!114 &11400000
MonoBehaviour:
  m_ObjectHideFlags: 0
  m_CorrespondingSourceObject: {fileID: 0}
  m_PrefabInstance: {fileID: 0}
  m_PrefabAsset: {fileID: 0}
  m_GameObject: {fileID: 0}
  m_Enabled: 1
  m_EditorHideFlags: 0
  m_Script: {fileID: 11500000, guid:
1b9111171376bdd46851cb2895d4e505, type: 3}
  m_Name: IO Ethernet
  m_EditorClassIdentifier:
  PartName: Ethernet Port
  PartDescription: An Ethernet port on a motherboard allows
users to connect a computer
    to a local area network (LAN) using an Ethernet cable.
It enables internet connectivity
    and access to networked devices, and modern
motherboards often include built-in
    Ethernet ports with support for high-speed Ethernet
standards.
```

```
%YAML 1.1
%TAG !u! tag:unity3d.com,2011:
--- !u!114 &11400000
MonoBehaviour:
  m_ObjectHideFlags: 0
  m_CorrespondingSourceObject: {fileID: 0}
  m_PrefabInstance: {fileID: 0}
  m_PrefabAsset: {fileID: 0}
  m_GameObject: {fileID: 0}
  m_Enabled: 1
  m_EditorHideFlags: 0
  m_Script: {fileID: 11500000, guid:
1b9111171376bdd46851cb2895d4e505, type: 3}
  m_Name: IO USB 2.0
  m_EditorClassIdentifier:
  PartName: USB 2.0
  PartDescription: A USB 2.0 port on the back of a
motherboard is an older type of
    port used to connect USB 2.0 devices to a computer. It
offers data transfer rates
    of up to 480 Mbps and can be used to connect various
USB 2.0 devices, such as
    mice, keyboards, printers, external hard drives, and
flash drives, to the computer.
```

```
%YAML 1.1
%TAG !u! tag:unity3d.com,2011:
--- !u!114 &11400000
MonoBehaviour:
  m_ObjectHideFlags: 0
  m_CorrespondingSourceObject: {fileID: 0}
  m_PrefabInstance: {fileID: 0}
  m_PrefabAsset: {fileID: 0}
  m_GameObject: {fileID: 0}
  m_Enabled: 1
  m_EditorHideFlags: 0
```

```
  m_Script: {fileID: 11500000, guid:
1b9111171376bdd46851cb2895d4e505, type: 3}
  m_Name: IO USB 3.0
  m_EditorClassIdentifier:
  PartName: USB 3.0
  PartDescription: A USB 3.0 port on the back of a
motherboard offers faster data
    transfer rates than the older USB 2.0 standard, with
speeds of up to 5 Gbps.
    The USB 3.0 port is easily identifiable by its blue
color and is used to connect
    USB 3.0 devices, such as external hard drives and flash
drives, to the computer.
```

```
%YAML 1.1
%TAG !u! tag:unity3d.com,2011:
--- !u!114 &11400000
MonoBehaviour:
  m_ObjectHideFlags: 0
  m_CorrespondingSourceObject: {fileID: 0}
  m_PrefabInstance: {fileID: 0}
  m_PrefabAsset: {fileID: 0}
  m_GameObject: {fileID: 0}
  m_Enabled: 1
  m_EditorHideFlags: 0
  m_Script: {fileID: 11500000, guid:
1b9111171376bdd46851cb2895d4e505, type: 3}
  m_Name: IO USB 3.2
  m_EditorClassIdentifier:
  PartName: USB 3.2
  PartDescription: A USB 3.2 port on the back of a
motherboard is a newer type of
    port used to connect USB 3.2 devices to a computer. It
offers faster data transfer
    rates than USB 3.0, with speeds of up to 20 Gbps, and
can be used to connect
    various USB 3.2 devices, such as external hard drives
and flash drives, to the
    computer.
```

```
%YAML 1.1
%TAG !u! tag:unity3d.com,2011:
--- !u!114 &11400000
MonoBehaviour:
  m_ObjectHideFlags: 0
  m_CorrespondingSourceObject: {fileID: 0}
  m_PrefabInstance: {fileID: 0}
  m_PrefabAsset: {fileID: 0}
  m_GameObject: {fileID: 0}
  m_Enabled: 1
  m_EditorHideFlags: 0
  m_Script: {fileID: 11500000, guid:
1b9111171376bdd46851cb2895d4e505, type: 3}
  m_Name: M.2
  m_EditorClassIdentifier:
  PartName: M.2 Slot
  PartDescription: M.2 slot is a high-speed expansion slot
found on modern motherboards
    that allows for the direct connection of solid-state
drives (SSDs) and other
    high-performance devices. M.2 slots can support
different interfaces like PCIe
    and SATA and offer faster data transfer rates and
better system performance compared
    to traditional hard disk drives, making them ideal for
high-speed storage and
    high-performance computing applications.
```

```
%YAML 1.1
%TAG !u! tag:unity3d.com,2011:
--- !u!114 &11400000
MonoBehaviour:
  m_ObjectHideFlags: 0
  m_CorrespondingSourceObject: {fileID: 0}
  m_PrefabInstance: {fileID: 0}
  m_PrefabAsset: {fileID: 0}
  m_GameObject: {fileID: 0}
  m_Enabled: 1
  m_EditorHideFlags: 0
  m_Script: {fileID: 11500000, guid:
```

```
1b9111171376bdd46851cb2895d4e505, type: 3}
  m_Name: PCIEx1
  m_EditorClassIdentifier:
  PartName: PCIe x1 Slot
  PartDescription: A PCIe x1 slot is a type of expansion
slot found on a motherboard
    that is used to connect various peripheral devices to a
computer, such as network
    cards, sound cards, and wireless cards. The PCIe x1
slot provides one lane of
    bandwidth for data transfer and is typically smaller
than other PCIe slots on
    a motherboard.
```

```
%YAML 1.1
%TAG !u! tag:unity3d.com,2011:
--- !u!114 &11400000
MonoBehaviour:
  m_ObjectHideFlags: 0
  m_CorrespondingSourceObject: {fileID: 0}
  m_PrefabInstance: {fileID: 0}
  m_PrefabAsset: {fileID: 0}
  m_GameObject: {fileID: 0}
  m_Enabled: 1
  m_EditorHideFlags: 0
  m_Script: {fileID: 11500000, guid:
1b9111171376bdd46851cb2895d4e505, type: 3}
  m_Name: PCIEx16
  m_EditorClassIdentifier:
  PartName: PCIe x16 Slot
  PartDescription: A PCIe x16 slot on a motherboard is a
type of expansion slot that
    can be used to connect high-performance PCIe devices,
such as graphics cards,
    to a computer. PCIe x16 slots offer faster data
transfer rates and more bandwidth
    than older PCI and AGP slots, and they are typically
used for gaming, video editing,
    and other graphics-intensive applications that require
high-performance graphics
    processing.
```

```
%YAML 1.1
%TAG !u! tag:unity3d.com,2011:
--- !u!114 &11400000
MonoBehaviour:
  m_ObjectHideFlags: 0
  m_CorrespondingSourceObject: {fileID: 0}
  m_PrefabInstance: {fileID: 0}
  m_PrefabAsset: {fileID: 0}
  m_GameObject: {fileID: 0}
  m_Enabled: 1
  m_EditorHideFlags: 0
  m_Script: {fileID: 11500000, guid:
1b9111171376bdd46851cb2895d4e505, type: 3}
  m_Name: Pin Audio
  m_EditorClassIdentifier:
  PartName: Audio Pin Header
  PartDescription: An audio pin header on a motherboard is
a connector that provides
    an interface for connecting an audio cable from the
front panel of the computer
    case to the motherboard. It enables the front panel
audio jacks to be used for
    headphones, microphones, and other audio devices
without having to plug them
    into the audio ports on the back of the computer.
```

```
%YAML 1.1
%TAG !u! tag:unity3d.com,2011:
--- !u!114 &11400000
MonoBehaviour:
  m_ObjectHideFlags: 0
  m_CorrespondingSourceObject: {fileID: 0}
  m_PrefabInstance: {fileID: 0}
  m_PrefabAsset: {fileID: 0}
  m_GameObject: {fileID: 0}
  m_Enabled: 1
  m_EditorHideFlags: 0
  m_Script: {fileID: 11500000, guid:
```

```
1b9111171376bdd46851cb2895d4e505, type: 3}
  m_Name: Pin Chassis Intrusion
  m_EditorClassIdentifier:
  PartName: Chassiss Intrusion Pin Header
  PartDescription: The Chassis Intrusion Pin Header is a
connector on a motherboard
    that enables the motherboard to detect if the computer
case has been opened or
    tampered with. It allows the motherboard to monitor the
state of the pin and
    alert the user if the computer case has been opened or
the pin is no longer connected.
```

```
%YAML 1.1
%TAG !u! tag:unity3d.com,2011:
--- !u!114 &11400000
MonoBehaviour:
  m_ObjectHideFlags: 0
  m_CorrespondingSourceObject: {fileID: 0}
  m_PrefabInstance: {fileID: 0}
  m_PrefabAsset: {fileID: 0}
  m_GameObject: {fileID: 0}
  m_Enabled: 1
  m_EditorHideFlags: 0
  m_Script: {fileID: 11500000, guid:
1b9111171376bdd46851cb2895d4e505, type: 3}
  m_Name: Pin CIMOS Battery
  m_EditorClassIdentifier:
  PartName: CIMOS Battery Pin Header
  PartDescription: The CMOS battery pin header on a
motherboard is a connection point
    for a CMOS battery, which powers the CMOS memory that
stores the BIOS settings
    and system clock information. The CMOS battery pin
header allows for easy replacement
    of the battery when it runs out of power or fails,
ensuring that the BIOS settings
    and system clock remain accurate.
```

```
%YAML 1.1
%TAG !u! tag:unity3d.com,2011:
--- !u!114 &11400000
MonoBehaviour:
  m_ObjectHideFlags: 0
  m_CorrespondingSourceObject: {fileID: 0}
  m_PrefabInstance: {fileID: 0}
  m_PrefabAsset: {fileID: 0}
  m_GameObject: {fileID: 0}
  m_Enabled: 1
  m_EditorHideFlags: 0
  m_Script: {fileID: 11500000, guid:
1b9111171376bdd46851cb2895d4e505, type: 3}
  m_Name: Pin Front Panel
  m_EditorClassIdentifier:
  PartName: Front Panel Pin Header
  PartDescription: A front panel pin header on a
motherboard is a connector that
    allows cables from the front panel of a computer case
to be connected to the
    motherboard. It includes connections for buttons, LEDs,
and audio jacks, enabling
    control of the computer through the front panel.
```

```
%YAML 1.1
%TAG !u! tag:unity3d.com,2011:
--- !u!114 &11400000
MonoBehaviour:
  m_ObjectHideFlags: 0
  m_CorrespondingSourceObject: {fileID: 0}
  m_PrefabInstance: {fileID: 0}
  m_PrefabAsset: {fileID: 0}
  m_GameObject: {fileID: 0}
  m_Enabled: 1
  m_EditorHideFlags: 0
  m_Script: {fileID: 11500000, guid:
1b9111171376bdd46851cb2895d4e505, type: 3}
  m_Name: Pin Parallel
```

```
  m_EditorClassIdentifier:
  PartName: Parallel Port Connector
  PartDescription: A parallel port connector on a
motherboard is a 25-pin female
    connector that allows data transfer between the
motherboard and parallel devices
    like printers or scanners. It is an older technology
that has largely been replaced
    by USB ports.
```

```
%YAML 1.1
%TAG !u! tag:unity3d.com,2011:
--- !u!114 &11400000
MonoBehaviour:
  m_ObjectHideFlags: 0
  m_CorrespondingSourceObject: {fileID: 0}
  m_PrefabInstance: {fileID: 0}
  m_PrefabAsset: {fileID: 0}
  m_GameObject: {fileID: 0}
  m_Enabled: 1
  m_EditorHideFlags: 0
  m_Script: {fileID: 11500000, guid:
1b9111171376bdd46851cb2895d4e505, type: 3}
  m_Name: Pin RBG
  m_EditorClassIdentifier:
  PartName: RGB Pin Header
  PartDescription: An RGB pin header on a motherboard is a
connector that allows
    for the connection of RGB LED strips or other RGB
lighting devices to the motherboard.
    It allows for control of the lighting effects through
software or hardware settings,
    allowing users to customize the appearance of their
computer.
```

```
%YAML 1.1
%TAG !u! tag:unity3d.com,2011:
--- !u!114 &11400000
MonoBehaviour:
  m_ObjectHideFlags: 0
  m_CorrespondingSourceObject: {fileID: 0}
  m_PrefabInstance: {fileID: 0}
  m_PrefabAsset: {fileID: 0}
  m_GameObject: {fileID: 0}
  m_Enabled: 1
  m_EditorHideFlags: 0
  m_Script: {fileID: 11500000, guid:
1b9111171376bdd46851cb2895d4e505, type: 3}
  m_Name: Pin Serial
  m_EditorClassIdentifier:
  PartName: Serial Port Connector
  PartDescription: A serial port connector on a motherboard
is used for serial communication
    with devices like modems and printers. It is a D-sub
connector with 9 or 25 pins,
    and is usually found on older motherboards or
specialized hardware.
```


```
%YAML 1.1
%TAG !u! tag:unity3d.com,2011:
--- !u!114 &11400000
MonoBehaviour:
  m_ObjectHideFlags: 0
  m_CorrespondingSourceObject: {fileID: 0}
  m_PrefabInstance: {fileID: 0}
  m_PrefabAsset: {fileID: 0}
  m_GameObject: {fileID: 0}
  m_Enabled: 1
  m_EditorHideFlags: 0
  m_Script: {fileID: 11500000, guid:
1b9111171376bdd46851cb2895d4e505, type: 3}
```

```
  m_Name: Pin TPM
  m_EditorClassIdentifier:
  PartName: TPM Pin Header
  PartDescription: A TPM (Trusted Platform Module) pin
header on a motherboard is
    a connector that allows for the installation of a TPM
module. A TPM module is
    a chip that provides hardware-based security features,
such as encryption and
    secure boot, and is used to enhance the security of the
computer.
```

```
%YAML 1.1
%TAG !u! tag:unity3d.com,2011:
--- !u!114 &11400000
MonoBehaviour:
  m_ObjectHideFlags: 0
  m_CorrespondingSourceObject: {fileID: 0}
  m_PrefabInstance: {fileID: 0}
  m_PrefabAsset: {fileID: 0}
  m_GameObject: {fileID: 0}
  m_Enabled: 1
  m_EditorHideFlags: 0
  m_Script: {fileID: 11500000, guid:
1b9111171376bdd46851cb2895d4e505, type: 3}
  m_Name: Pin USB 2.0
  m_EditorClassIdentifier:
  PartName: USB 2.0 Pin Header
  PartDescription: A USB 2.0 pin header on a motherboard is
a connector that allows
    for the installation of USB 2.0 ports on the computer
case. It provides a way
    to connect the USB 2.0 ports on the front panel of the
case to the motherboard,
    allowing for easy access to USB ports on the front of
the computer.
```

```
%YAML 1.1
%TAG !u! tag:unity3d.com,2011:
--- !u!114 &11400000
MonoBehaviour:
  m_ObjectHideFlags: 0
  m_CorrespondingSourceObject: {fileID: 0}
  m_PrefabInstance: {fileID: 0}
  m_PrefabAsset: {fileID: 0}
  m_GameObject: {fileID: 0}
  m_Enabled: 1
  m_EditorHideFlags: 0
  m_Script: {fileID: 11500000, guid:
1b9111171376bdd46851cb2895d4e505, type: 3}
  m_Name: Pin USB 3.0
  m_EditorClassIdentifier:
  PartName: USB 3.0 Pin Header
  PartDescription: A USB 3.0 pin header on a motherboard is
a connector that provides
    an internal USB 3.0 port. This allows for the
connection of USB 3.0 devices to
    the motherboard, such as internal card readers or front
panel USB 3.0 ports on
    the computer case.
```

```
%YAML 1.1
%TAG !u! tag:unity3d.com,2011:
--- !u!114 &11400000
MonoBehaviour:
  m_ObjectHideFlags: 0
  m_CorrespondingSourceObject: {fileID: 0}
  m_PrefabInstance: {fileID: 0}
  m_PrefabAsset: {fileID: 0}
  m_GameObject: {fileID: 0}
  m_Enabled: 1
  m_EditorHideFlags: 0
  m_Script: {fileID: 11500000, guid:
1b9111171376bdd46851cb2895d4e505, type: 3}
  m_Name: RAM
```

```
  m_EditorClassIdentifier:
  PartName: Random Access Memory
  PartDescription: "\rRAM, or Random Access Memory, allows
for quick access to data
    and is used to temporarily store active data for the
CPU. It is faster than permanent
    storage devices and improves a computer's performance.
More RAM can enhance the
    overall speed and responsiveness of a system, making it
an important consideration
    when upgrading or purchasing a computer."
```

```
%YAML 1.1
%TAG !u! tag:unity3d.com,2011:
--- !u!114 &11400000
MonoBehaviour:
  m_ObjectHideFlags: 0
  m_CorrespondingSourceObject: {fileID: 0}
  m_PrefabInstance: {fileID: 0}
  m_PrefabAsset: {fileID: 0}
  m_GameObject: {fileID: 0}
  m_Enabled: 1
  m_EditorHideFlags: 0
  m_Script: {fileID: 11500000, guid:
1b9111171376bdd46851cb2895d4e505, type: 3}
  m_Name: RAM Slot
  m_EditorClassIdentifier:
  PartName: RAM Slot
  PartDescription: A RAM slot on a motherboard is a
connector where the RAM (Random
    Access Memory) modules are inserted. RAM is a type of
memory that allows the
    computer to temporarily store and access data quickly.
The number and type of
    RAM slots on a motherboard can vary, and it determines
the maximum amount and
    speed of RAM that can be installed in the computer.
```

```
%YAML 1.1
%TAG !u! tag:unity3d.com,2011:
--- !u!114 &11400000
MonoBehaviour:
  m_ObjectHideFlags: 0
  m_CorrespondingSourceObject: {fileID: 0}
  m_PrefabInstance: {fileID: 0}
  m_PrefabAsset: {fileID: 0}
  m_GameObject: {fileID: 0}
  m_Enabled: 1
  m_EditorHideFlags: 0
  m_Script: {fileID: 11500000, guid:
1b9111171376bdd46851cb2895d4e505, type: 3}
  m_Name: Sata Ports
  m_EditorClassIdentifier:
  PartName: SATA Ports
  PartDescription: SATA ports on a motherboard are
connectors that allow you to connect
    storage devices to the motherboard for data transfer.
Most modern motherboards
    have several SATA ports available for connecting
multiple storage devices.
```

```
%YAML 1.1
%TAG !u! tag:unity3d.com,2011:
--- !u!114 &11400000
MonoBehaviour:
  m_ObjectHideFlags: 0
  m_CorrespondingSourceObject: {fileID: 0}
  m_PrefabInstance: {fileID: 0}
  m_PrefabAsset: {fileID: 0}
  m_GameObject: {fileID: 0}
  m_Enabled: 1
  m_EditorHideFlags: 0
  m_Script: {fileID: 11500000, guid:
1b9111171376bdd46851cb2895d4e505, type: 3}
  m_Name: SSD
  m_EditorClassIdentifier:
```

```
  PartName: SSD
  PartDescription: SSD is a type of storage device that
uses flash memory instead
    of mechanical parts. It's faster, more durable, and
more reliable than traditional
    hard drives, making it a popular choice for use in
laptops, desktops, and servers.
    SSDs come in different sizes and can store a lot of
data.
```

```
%YAML 1.1
%TAG !u! tag:unity3d.com,2011:
--- !u!114 &11400000
MonoBehaviour:
  m_ObjectHideFlags: 0
  m_CorrespondingSourceObject: {fileID: 0}
  m_PrefabInstance: {fileID: 0}
  m_PrefabAsset: {fileID: 0}
  m_GameObject: {fileID: 0}
  m_Enabled: 1
  m_EditorHideFlags: 0
  m_Script: {fileID: 11500000, guid:
1b9111171376bdd46851cb2895d4e505, type: 3}
  m_Name: VRM
  m_EditorClassIdentifier:
  PartName: 'Voltage Regulator Module '
  PartDescription: VRM refers to a group of electronic
components on a motherboard
    that regulate the voltage supplied to the CPU. The VRM
helps ensure stable and
    consistent power delivery to the CPU, which is
important for the performance
    and longevity of the processor.
```

CURRICULUM VITAE

Personal:

Name: Clarence Lovely D. Panizales

Birthdate: July 4, 1998

Place Of Birth: Lincud, Dingle, Iloilo

Weight: 42 kg Height: 150 cm

Marital Status: Single

Nationality: Filipino

Address:

Email Address: lovelyclarencep@gmail.com

Mobile Phone: 09287301381

Permanent Address: Angeles St., San Enrique, Iloilo

Education:

Primary: Lincud Elementary School (2010)

Secondary: Dingle National High School (2015)

Tertiary: Interface Computer College Inc. Iloilo (2023)

Personal:

Name: Lindsay Ruivivar

Birthdate: July 13, 2001

Place Of Birth: Cabatuan, Iloilo

Weight: 51 kg Height: 165 cm

Marital Status: Single

Nationality: Filipino

Address:

Email Address: lindsruivivar09@gmail.com

Mobile Phone: 09196899683

Permanent Address: Bacan, Cabatuan, Iloilo

Education:

Primary: Cabatuan Central Elementary School (2013)

Secondary: Western Institute of Technology (2019)

Tertiary: Interface Computer College Inc. Iloilo (2023)

Personal:

Name: Jhon Prietse D. Tacaisan

Birthdate: December 11, 2000

Place Of Birth: Jordan, Guimaras

Weight: 55 kg Height: 166 cm

Marital Status: Single

Nationality: Filipino

Address:

Email Address: prietse.personal+ICCAR@gmail.com

Mobile Phone: 09384239896

Permanent Address: Bugnay, Jordan, Guimaras

Education:

Primary: Surigao City Central School (2013)

Secondary: Guimaras State College (2019)

Tertiary: Interface Computer College Inc. Iloilo (2023)