Customizable VR Design of Friends Interaction for Social Story

BY

TEE ZI JUN

A REPORT

SUBMITTED TO

Universiti Tunku Abdul Rahman

in partial fulfillment of the requirements

for the degree of

BACHELOR OF COMPUTER SCIENCE (HONOURS)

Faculty of Information and Communication Technology

(Kampar Campus)

JANUARY 2024

UNIVERSITI TUNKU ABDUL RAHMAN

REPORT STATUS DECLARATION FORM

Title: Customizable VR Design of Friends Interaction for Social Story

Academic Session: JANUARY 2024

I TEE ZI JUN

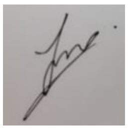
(CAPITAL LETTER)

declare that I allow this Final Year Project Report to be kept in

Universiti Tunku Abdul Rahman Library subject to the regulations as follows:

1. The dissertation is a property of the Library.
2. The Library is allowed to make copies of this dissertation for academic purposes.

Verified by,



(Author's signature)

(Supervisor's signature)

Address:
8, Jalan Dato Kaya Kecil,
Taman Mutiara,
42200 Klang, Selangor.

Ts Dr Goh Hock Guan
Supervisor's name

Date: 15/04/2024

Date: 15/04/2024

Universiti Tunku Abdul Rahman			
Form Title : **Sample of Submission Sheet for FYP/Dissertation/Thesis**			
Form Number: **FM-IAD-004**	Rev No.: **0**	Effective Date: **21 JUNE 2011**	Page No.: **1 of 1**

FACULTY OF INFORMATION AND COMMUNICATION TECHNOLOGY

UNIVERSITI TUNKU ABDUL RAHMAN

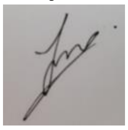
Date: 15/04/2024

SUBMISSION OF FINAL YEAR PROJECT

It is hereby certified that **TEE ZI JUN** (ID No: **20ACB00978**) has completed this final year project entitled "Customizable VR Design of Friends Interaction for Social Story" under the supervision of Ts Dr Goh Hock Guan (Supervisor) from the Department of Computer and Communication Technology , Faculty of Information and Communication Technology.

I understand that University will upload softcopy of my final year project in pdf format into UTAR Institutional Repository, which may be made accessible to UTAR community and public.

Yours truly,

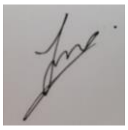


(*TEE ZI JUN*)

*Delete whichever not applicable

DECLARATION OF ORIGINALITY

I declare that this report entitled "**Customizable VR Design of Friends Interaction for Social Story**" is my own work except as cited in the references. The report has not been accepted for any degree and is not being submitted concurrently in candidature for any degree or other award.



Signature :

Name : TEE ZI JUN

Date : 15/04/2024

ACKNOWLEDGEMENTS

I would like to express my sincere thanks and appreciation to my supervisor, Ts Dr Goh Hock Guan who has given me this amazing opportunity to involve in an VR environment development project. I believe that this project will enhance my career profile and pave the way for a better future. Moreover, my supervisor has offered valuable guidance and insights throughout this project. Once again, a million thanks to my supervisor.

ABSTRACT

The primary goal of this project is to construct an immersive and customizable virtual reality (VR) experience to empower autistic primary school students to rehearse social interactions involved in forming new friendships, within a simulated classroom environment. The customized VR setting provides opportunities for autistic students to safely hone their social interactions including self-introductions, initiating dialogues, deciphering social cues, and building friendship. Parents can tailor class background, classroom objects, language, sound setting, and scenarios to match with their child's actual classroom contexts and preferences. By harnessing VR's interactive features, this project seeks to refine targeted interpersonal skills and elevate social communication adeptness in autistic students. Expert psychologists will assess the solution's viability, usability, engagement, and learning potential. Their evaluations will provide insights into effectively using VR technology to enhance the social skillset of autistic students. This project constitutes a component of a larger VR project with a team of three working together in software development, software testing, and VR content creation. Each member is tasked to construct one scenario that could occur during school. This project scope is to assist autistic primary school students in forming friendships through this customizable VR application.

Furthermore, the VR application has been successfully developed and deployed on an Android device. Unity is the major software used to create the customizable VR application with Google Cardboard VR framework, while Blender is used to create the avatar models and animations. The C# programming language is used to complete the project's scripting, and Microsoft Visual Studio is the development environment of choice.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS

VR	Virtual Reality
ASD	Autism Spectrum Disorder
CAVE	Cave Automatic Virtual Environment
PEP-3	Psychoeducational Profile, Third Edition
HMD	head-mounted display
LFA	Low Functioning Autism
URP	Universal Render Pipeline
UE5	Unreal Engine 5
RAD	Rapid Application Development
AI	Artificial Intelligence
NPC	Non-playable Character

CHAPTER 1

Introduction

1.1 Problem Statement and Motivation

The absence of audio voiceovers and background music in the social story hinders the development of an immersive and realistic learning environment. Sound aspects are critical for creating a feeling of presence and atmosphere in the virtual environments, allowing autistic students to fully participate with the simulated scenarios and social interactions. Without proper background music and audio voiceovers, the virtual experience may appear lifeless and separated from reality. This may weaken the authenticity of the experience and reduce the effectiveness of the learning process. Furthermore, the lack of audio voiceovers could limit auditory learners' ability to completely absorb and recall the social story material. In short, the project should integrate the audio voiceovers and background music to make the virtual environments more lifelike and immersive.

Another problem is the lack of interaction and involvement in the social story greatly restricts autistic students' ability to gain and retain skills [1]. Social interactions are dynamic and multifaceted, incorporating verbal and nonverbal signs, body language, and appropriate replies. Without interactive elements, the social story may appear static and one-dimensional, missing the details of real-life social settings. Interactive scenarios in which autistic children may actively engage, make decisions, and get feedback based on their actions are critical for improving their knowledge and application of social skills. In the absence of such interactions and engagements, students may lose interest and motivation, thereby affecting the entire learning experience and the effectiveness of the VR application. Therefore, the project must create a more engaging and individualized social story that encompasses all areas of school life and the building of friendships.

The necessity to address the drawbacks and shortcomings of the existing VR application for enhancing friendship skills in students with ASD is the motivation behind this project [1]. The absence of audio elements and the lack of interaction in the social story of the present VR solution may not meet the unique requirements and preferences of autistic students. As a result, the purpose of this project is to create and assess a VR environment that can be customized to meet the needs and preferences of parents whose students has ASD. In this approach, the VR intervention may teach social skills in a more relevant and effective fashion, which can enhance the engagement and quality of life for students with ASD.

1.2 Project Objectives

The main objective of this project is to create an immersive, accessible, and customized VR application using the Unity engine and Google Cardboard VR. The VR application will provide entertaining, repeatable experiences for primary school students with ASD to practice social skills and challenging situations related to initiating friendships in a simulated school environment tailored to their individual needs and preferences.

One of the sub-objectives of this project is to develop a VR application that allows parents to customize the VR environment, language, and social stories. This VR application will provide parents with an easy-to-use interface that will enable them to control over many features of the simulated environment. This includes customizing the background settings, thereby increasing their child's motivation and involvement in the VR environment.

Another sub-objective of this project is to create social stories that are relevant to each autistic primary student's experiences and needs. Impactful social stories will be created by understanding the autistic primary student’s challenges. These social stories will be tailored to each autistic primary student's specific needs and interests. For example, if a primary student has difficulty establishing friends, a story on sharing, taking turns, and listening could be appropriate. Making the story more realistic increases engagement and teaches important social skills. These stories, which are based on real-life circumstances, seek to deliver important lessons, so greatly contributing to the autistic primary student's social development journey.

This project's novelty is its use of VR technology to provide more tailored, interesting, and engaging therapies for enhancing friend interaction skills in primary students with ASD. Additionally, the results will develop knowledge of how VR customisation may improve social learning outcomes depending on unique requirements and objectives.

1.3 Project Scope

This project aims to develop an immersive and customizable VR software solution that allows autistic primary school students to practice social interactions and situations related to initiating friendships within a simulated classroom environment. The proposed solution is a VR learning tool that will provide opportunities for students to rehearse key social skills such as introducing themselves, starting conversations, interpreting social cues, and building connections with virtual classmates. The product will incorporate customized virtual classroom settings, language settings, and social scenarios that can be tailored to each student's educational context, needs, and goals. At the end of this project, the deliverable will be a working prototype of the customizable VR classroom friendship simulation application capable of being accessed through affordable Google Cardboard headsets to provide an engaging and personalized social skill learning experience.

1.3 Expected Contributions from the Project

This project on customizable VR for friend interaction skills practice can make valuable contributions. Firstly, it involves developing innovative customized VR technology for personalized social learning in ASD. Secondly, it can provide insight on the usability, acceptability, and feasibility of this VR project among children with ASD. Thirdly, it enables the identification of best practices in tailored VR content design matched to student profiles. Fourthly, this project utilizes customized VR interactions to enhance traditional teaching approaches like social stories. Finally, this project can advance the understanding of VR personalization advantages for improving social functioning in students with diverse needs across the autism spectrum.

1.4 Organization of the Report

The details and background of this project are shown in the current chapter (Chapter 1). Next, Chapter 2 shows the reviews of related technologies and existing systems, along with a comparison between them. Then, the following chapter will discuss the system methodology, which includes selection of the system development model, system requirement, functional requirement, expected challenges, system testing and performance along with project timeline. After that, Chapter 4 is going to explain the system design in detail. On Chapter 5, the setup and configuration of software will be shown together with the system operation. The evaluation and discussion of the software will be explained in Chapter 6. Lastly, Chapter 7 will discuss the conclusion and recommendation of this project.

1.5 Concluding Remark

This chapter defines the problem statement and motivation of the VR project. After that, the project objectives and project scope will be discussed. The chapter will further analyse the contributions from the VR project. Lastly, the overall organization of the report is shown in this chapter.

CHAPTER 2

Literature Review

2.1 Review of the Technologies

2.1.1 Unity

Unity is a popular game engine and development platform that allows developers to create VR experiences for a variety of platforms and devices. Developers may use Unity's capabilities and toolset to create immersive and powerful VR apps.

The Universal Render Pipeline (URP), a versatile and adaptable graphics pipeline that enables developers to achieve outstanding graphics quality and performance across various kinds of hardware requirements, is one of Unity's primary features for VR development. In addition to supporting VR rendering, URP offers tools like lighting settings, shader graphs, and post-processing to improve the visual realism and visual integrity of VR environments [2].

Another feature of Unity for VR development is the XR Interaction Toolkit, which is a high-level, component-based interaction framework for generating VR and AR experiences. The XR Interaction Toolkit provides a framework for generating 3D and UI interactions using input events, cross-platform XR controller inputs, haptics, visual feedback, basic canvas UI, and other sources. On top of that, developers may offer a variety of movement choices for users, including teleportation, continuous moving, and snap turning, thanks to the XR Interaction Toolkit's customizable locomotion system.

Along with these tools, Unity also provides visual scripting, which eliminates the need to write code entirely in favour of generating scripting logic using visual, drag-and-drop graphs. Unity's visual scripting feature facilitates more fluid cooperation between designers, programmers, and artists for quicker iterations and prototyping, which speeds up the creation of virtual reality.

Finally, Unity is compatible with several VR headsets and platforms, including Meta Quest, PlayStation VR, SteamVR, and others. Unity offers plug-ins and platform-specific packages that let developers utilize all of the VR devices' built-in functionality. For instance, the Oculus Integration package offers resources and tools for creating virtual reality apps for Oculus hardware, including the Oculus Quest and Rift.

2.1.2 Unreal Engine 5

The most recent iteration of Epic Games' well-known gaming engine is called Unreal Engine 5 (UE5). Nanite and Lumen, two ground-breaking features included in UE5, allow for high-end VR experiences.

Nanite is a virtualized geometry system that enables developers to design and render enormous volumes of complex geometry while minimizing performance overhead. With billions of polygons handled in every frame, Nanite enables VR at unparalleled levels of realism and immersion. Nanite will stream and scale the geometry data depending on the resolution and position of the camera. Besides that, collision detection, degree of detail, and instancing are supported by Nanite [3].

Lumen is a completely dynamic global illumination system that responds in real-time to changes in lighting and geometry. Lumen streamlines processes and saves time by keeping away from the need to place light probes or bake lightmaps. Complex lighting conditions including the sun, sky, reflections, shadows, and translucent materials can all be dealt with Lumen. Other than that, Lumen also adjusts to the resolution and field of vision of the VR headset, assuring consistent quality and performance [4].

Numerous VR headsets, including Oculus Quest, Rift, Valve Index, HTC Vive, and Windows Mixed Reality, are compatible with UE5's Nanite and Lumen. OpenXR, a multi-company standard for VR and AR development, is also supported by UE5. Without the need for platform-specific checks or calls, developers can leverage the same code and logic across several platforms and devices thanks to OpenXR. In addition, UE5 has a VR mode in the editor that allows developers to construct and test their VR environments in VR using intuitive tools and interactions [5].

2.1.3 Meta Quest 3

The company Meta, formerly known as Facebook, has unveiled a new virtual and mixed-reality headgear called the Meta Quest 3. It is a successor to the 2020-released Meta Quest 2. Compared to the Quest 2, the Meta Quest 3 has quite a few upgrades.

Firstly, the resolution of the Meta Quest 3 is 50% larger than that of the Quest 2, with each eye having 2560 × 2560 pixels as opposed to 1832 x 1920 pixels. This leads to clearer and more lifelike VR and MR images. In addition, the Qualcomm Snapdragon XR3 platform, which is developed exclusively for VR and MR devices, powers the Meta Quest 3. It provides a faster CPU, GPU, and AI engine than the Snapdragon XR2 platform seen in the Quest 2. This implies that VR and MR experiences will be smoother and more immersive [6].

A new technology called Meta Reality, which smoothly merges the digital and real worlds, is introduced in The Meta Quest 3. Meta Reality maps the user's surroundings and overlays realistic 3D objects and effects using sophisticated sensors, cameras, and software. The user can utilize controls, their voice, or their hands to interact with these items and effects. Social presence is another feature of meta-reality that lets users see and interact with other users in VR and MR.

On top of that, the Meta Quest 3 has pancake optics, a novel form of lens that decreases the headset's thickness and weight. Compared to the Quest 2, the Meta Quest 3 is 30% thinner and 20% lighter. It is also more ergonomic, with better straps, cushioning, and ventilation. The Meta Quest 3 is made to be worn for extended periods without causing weariness or pain.

Moreover, the Meta Quest 3 provides players with access to a variety of VR and MR games and apps, including Beat Saber, Superhot VR, Horizon Worlds, and others. The Meta Quest 3 is compatible with the Meta Link connection, which lets users connect the headset to a PC and play PC VR games and apps.

Figure 2.1.3.1 Meta Quest 3

2.1.4 Valve Index

The Valve Index is a high-end VR headgear created by Valve that provides an exceptional VR experience for PC gamers. It has a broad field of view of 130 degrees, a dual LCD with a resolution of 1440x1600 per eye, a refresh rate of up to 144 Hz, and two cutting-edge controllers that can follow each finger's movement. Along with these features, the headset has an adjustable and comfortable design, a built-in speaker system that produces spatial sounds, and a front compartment that may hold additional sensors or modules [7].

The Valve Index's major selling point is its controllers, often known as Knuckles. Because of the ergonomic design and easy functionality of these controllers, users may interact with virtual things naturally. The controllers can detect the location and pressure of each finger, allowing motions and expressions that current VR controllers cannot. Apart from that, the controllers have a grip button, a trigger, a joystick, a trackpad, and a system button, offering plenty of input choices for various games and apps.

The biggest and most varied VR content platform on PCs is SteamVR, and it works with the Valve Index. SteamVR users get access to thousands of VR games, applications, and experiences, ranging from AAA blockbusters such as Half-Life: Alyx and No Man's Sky to indie gems like Beat Saber and Superhot VR. To enable users to create and share their own virtual reality experiences, SteamVR now enables user-generated content, including modifications, maps, and personalized avatars.

Figure 2.1.4.1 Valve Index

2.1.5 Summary of the Technologies Review

Technologies	Features
Unity	• Universal Render Pipeline (URP) that provides outstanding graphics quality and performance across various kinds of hardware requirements. • XR Interaction Toolkit, which is a high-level, component-based interaction framework for generating VR and AR experiences. • Visual scripting to generate scripting logic using visual, drag-and-drop graphs. • Compatible with several VR headsets and platforms, including Meta Quest, PlayStation VR, SteamVR, and others.
Unreal Engine 5	• Nanite is a virtualized geometry system that enables developers to design and render enormous volumes of complex geometry while minimizing performance overhead. • Lumen is a completely dynamic global illumination system that responds in real-time to changes in lighting and geometry. • Compatible with VR headsets, including Oculus Quest, Rift, Valve Index, HTC Vive, and Windows Mixed Reality. • OpenXR is supported.
Meta Quest 3	• Each eye having 2560 × 2560 pixels of resolution. • Powered by Qualcomm Snapdragon XR3 platform. • Integrated with Meta Reality, which smoothly merges the digital and real worlds. • Utilized pancake optics, a novel form of lens that decreases the headset's thickness and weight. • More ergonomic with better straps, cushioning, and ventilation.

	• Access to a variety of VR and MR games and apps. • Compatible with the Meta Link connection, which connects the headset to a PC and play PC VR games and apps.
Valve Index	• Has a dual LCD with a resolution of 1440x1600 per eye, field of view of 130 degrees and a refresh rate of up to 144 Hz. • Has adjustable and comfortable design. • Integrated with a built-in speaker system that produces spatial sounds. • Comes with a front compartment that may hold additional sensors or modules. • Implemented ergonomic design controllers called Knuckles that provide easy functionality to enable users interact with virtual things naturally. • Primarily compatible with SteamVR.

Table 2.1.5.1 Summary of the Technologies Review

2.2 Review of the Existing Systems/Applications

2.2.1 Using Virtual Reality to Train Emotional and Social Skills in Children with Autism Spectrum Disorder

Yuan et al. 2018 [8] at the AIM Tech Centre in Hong Kong developed a VR training program to assist children with ASD. The program utilizes an immersive four-sided Cave Automatic Virtual Environment (CAVE) to deliver interactive VR scenarios. The training takes place in small groups of 3-4 children. Before entering the CAVE, they review previously learned skills and concepts. Each child then navigates the VR environment individually with guidance from a trainer. Six VR scenarios were designed based on real-life situations of primary school students in Hong Kong. These scenarios aim to teach social, emotional, and safety skills. The CAVE system surrounds the child with virtual environments projected on four walls and the floor. Motion sensors track the child's interactions. In the one-hour VR session, the trainer provides support as needed. Briefing and debriefing frame the experience to prepare children and generalize skills.

Primary school-aged children diagnosed with ASD were eligible. The majority were boys with a mean age of 106 months. Children with intellectual disabilities were excluded. The Psychoeducational Profile, Third Edition (PEP-3), assessed emotion expression, regulation, and social interaction skills before and after training. Qualitative feedback was also gathered. The results showed significant improvements in emotion expression, regulation, social interaction, and adaptation after training compared to before. Qualitative feedback was positive, with increased socialization reported.

Given the promising results, the researchers started investigating more portable head-mounted displays (HMDs) that can be set up in schools to reach more children. Apart from social-emotional skills, VR could have broader applications such as anxiety relief or stigma education. Mobile VR increases convenience, but therapist guidance remains essential.

In summary, this immersive CAVE-based VR program successfully taught social and emotional skills to children with ASD. Both quantitative data and qualitative feedback showed improvements. While accessibility is limited, findings demonstrate VR's potential for building critical skills in safe and controlled environments. Ongoing efforts aim to transition the training to more widely available HMD platforms while retaining therapist support and real-world application. This research highlights VR as a promising tool for enhancing independence in ASD.

2.2.1.1 Strengths

The immersive four-sided CAVE system offers a high degree of fidelity and interaction for virtual training scenarios. Motion tracking and projection on all sides immerse children in simulated environments. The VR training resulted in significant quantitative gains in social, emotional, and adaptation skills. Qualitative feedback from parents and teachers also indicated positive real-world improvements. The briefing and debriefing framework helps generalize skills learned in VR to everyday situations.

2.2.1.2 Weakness

The CAVE platform demands significant dedicated space and involves high implementation and maintenance costs. This diminishes accessibility for numerous schools and organizations. The VR scenarios might lack long-term retention and sufficient transferability beyond the virtual environments. The training necessitates continuous therapist guidance, thereby raising personnel expenses. Participation was also restricted due to scheduling constraints for families.

2.2.1.3 Recommendation

To improve accessibility, the training could transition to more portable and affordable head-mounted VR systems. Low-cost mobile VR solutions could also be explored while balancing immersion. For retention and transfer, VR scenarios could incorporate more variable situations and realistic stimuli. Booster training sessions could help sustain gains over time. Therapist support could be supplemented with interactive virtual assistants to reduce personnel costs. To address scheduling issues, the VR training may need to be offered outside of school hours.

2.2.2 Developing an Immersive Virtual Reality Training System to Enrich Social Interaction and Communication Skills for Children with Autism Spectrum Disorder

Alimanova et al. 2022 [9 developed a VR training system to improve social interaction and communication skills in children diagnosed with ASD. The system utilizes VR technology, including a VR headset, camera, and microphone, to immerse autistic children in well-designed and realistic 3D environments. Within these environments, the children can practice and rehearse social situations and communication abilities.

The proposed system would employ real-time analysis of facial expressions and speech to assess the autistic child's emotional state and reactions in the moment. This quantified feedback would assist therapists in measuring the child's progress and milestones more effectively. A significant advantage of this system is that children can repeat lessons and continue practicing social skills in an engaging VR setting.

In the VR session, the autistic child engages with a virtual animated farmer character. The character poses questions and offers responsive feedback. The farmer's behaviours and reactions adjust based on the child's emotions, facial expressions, and verbal actions. Both verbal and non-verbal behaviours of the child are recorded for subsequent analysis and assessment. The VR experience is designed as an engaging game to promote children's practice and participation in social interactions.

The 3D visual components were constructed using Autodesk Maya software. These visuals were then integrated into the Unity platform, responsible for managing graphics, animations, physics simulations, and audio effects. Unity's built-in facial and voice recognition capabilities enabled real-time analysis of the child's emotions and speech. Fully immersive VR, complete with precise motion tracking, was achieved through Oculus Quest 2 headsets. The selection of Oculus Quest 2 was based on its superior accuracy, portability, display resolution, and reasonable cost compared to alternatives. Its standalone design eliminates the need for cables or external computing hardware. The battery life was sufficient for short therapy sessions.

The initial results indicated that autistic children were actively engaged with and intrigued by the VR environment and interactions. The system offered a high level of interactivity within a secure virtual space for practicing essential social skills. During the Covid-19 pandemic, individuals from remote areas have the chance to pursue therapy from their homes. The Oculus Quest 2 VR headset enables multiple users to connect to the training system. Psychotherapists can conduct therapy by linking to the system as virtual characters.

In summary, the suggested VR training system employs immersive technology, facial and speech analysis to enhance social communication skills in autistic children. While further research is required, initial testing yielded positive results, and progressive development persists. The system holds substantial potential for engaging, personalized social skills training, adapted to each child's requirements and advancement.

2.2.2.1 Strengths

One major strength of this system is the real-time integration of facial and voice analytics to assess the child's emotional responses while immersed in the VR training. This quantitative data gives therapists valuable feedback they can use to measure progress over time. The VR session is in the form of a game, which makes the experience enjoyable and promotes active engagement from the child. Furthermore, the VR technology offers convenient and interactive therapy sessions even in remote areas.

2.2.2.2 Weakness

A limitation of the system is that the virtual scenarios may not simulate real-world social situations. Thus, the communication skills practiced in VR sessions may not be applicable. Another weakness is that the facial and speech recognition are not robust enough to accurately interpret the child's emotions and meanings. The Oculus Quest 2 headset is reliant on battery life, which could limit session duration. Furthermore, the VR system hardware and software requirements are fairly expensive.

2.2.2.3 Recommendation

The virtual scenarios could be made more realistic through interactive avatars and environments to better simulate real-world social situations. Emotion and speech recognition could be improved using AI trained on child facial and vocal data. The headset could be corded to remove battery limitations. Lower-cost headset and software options could improve affordability. Cost-effectiveness can also be improved by optimizing VR software and considering financial support strategies such as subsidies.

2.2.3 Development of Virtual Reality Content to Improve Social Skills in Children with Low Function Autism

Universitas Negeri Malang. Fak… et al. n.d. [10] utilized the Borg and Gall development model to design and test a VR system aimed at helping develop social skills in children with Low Functioning Autism (LFA). The VR system's hardware comprised a Bobo VR head-mounted display, Oppo A12 smartphones, and a computer. The smartphones were configured as VR displays and controllers by installing an application, with the computer serving as an intermediary connection. Once installed and set up, the smartphones were able to mirror their VR display onto the computer screen.

The VR system was designed to simulate the scenario of making purchases in a school cafeteria. The virtual environment and avatars closely resembled real world aesthetics and functionality in order to enhance the transferability of acquired social skills. Familiar snack options were included for quicker adaptation. Each child had VR sessions lasting 10 to 15 minutes, based on their comprehension. The first trial familiarized children with VR, while later trials included specific instructions to evaluate skill development.

The results showed that 71.4% of children initially displayed apprehension or fear when introduced to the VR headset. In contrast, 57.1% expressed wonder upon encountering VR visuals, having no prior exposure to the medium. However, this initial shock gradually diminished as children recognized virtual objects corresponding to familiar real-world items. Post-simulation fatigue was noted in 71.4% of children. While positive responses were brought out for objects and colours, adhering to instructed activities proved challenging, with only 28.6% of children responding accurately and promptly. On average, teachers rated the VR content as 82.9% valid, confirming its strong validity. Nonetheless, concerns emerged about visual obstruction due to interface limitations and minor physical discomfort. Furthermore, the virtual shopping process highlighted the need for improvements to address LFA-specific issues related to central coherence and executive function.

In summary, the VR system and content, though still an early effort, showed reasonably effectiveness for low functioning autistic children, and high validity as rated by experienced teachers. The use of VR displays strong potential in enabling the development of vital social skills in LFA children to empower them to handle daily activities. However, limitations stemming from challenges with social interaction, attention spans, language abilities and communication were clearly evident in the test responses, necessitating design refinements. While simplicity, avatars and real-world environments help, overcoming core deficiencies related to central coherence and executive functions remain an open problem. Further research to address these areas could unlock VR's full promise for social adaptation in low functioning autism.

2.2.3.1 Strengths

It utilizes affordable and accessible VR hardware like smartphones and headsets, allowing for widespread use. The virtual environment and objects are readable and recognizable for users on the spectrum. The overall setup process is straightforward, and the real-world application of shopping provides a practical learning opportunity.

2.2.3.1 Weakness

Many users experienced initial anxiety and difficulty following multi-step instructions, indicating a need to simplify direction-following. The VR content also requires modifications to fully support LFA challenges with information processing and cognition. Some users reported physical discomfort from the headset as well. Visual occlusion of objects due to interface design was another issue.

2.2.3.1 Recommendation

Implementing gradual VR introduction with calming cues can lessen anxiety. Breaking down multi-step instructions into smaller segments, providing additional comprehension time, and using visual aids in following directions. Simplifying environments, offering decision guidance, and emphasizing cause-effect relationships assists those with limited cognition. Adjustable headset straps, breathable materials, speed toggling, and breaks enhance comfort. Decreasing clutter, repositioning essential objects, employing adaptive eye tracking views, and utilizing immersive 3D audio all work to minimize occlusion.

2.2.4 A Design of Multipurpose Virtual Reality Game for Children with Autism Spectrum Disorder

Institute of Electrical and El… n.d. [11] developed a VR game using Leap Motion hand tracking to assist children with ASD. The system hardware consists of a head-mounted display for immersive 3D visualization and a Leap Motion controller for capturing hand motions. The Leap Motion rig is attached to the head display, so the virtual hands remain visible as subjects look around. The VR game were developed in a 3D engine on a VR-ready PC. Four games were designed, targeting different shortcomings in ASD, such as focusing, filtering distractions, following rules and signs, and recognizing eye gaze.

The first game has children group the coloured balls appearing randomly into matching boxes to train sustained focus. Rewards like fruits are given for correct placements. The second game adds moving fish as distractions that children must ignore while grouping stationary fish by colour into boxes. Coins are provided as rewards. The third game has children navigate through an environment by following arrow signs that dictate directional movements to reach destinations. The fourth game has children identify objects being looked at by a virtual character based on eye gaze direction. Coins are awarded for correct selections.

The games went through iterative testing to refine the controls and environment. The ball sizes and graspable handles enabled pickup by the virtual hands. Convex mesh colliders enabled efficient collision detection for hand interactions. Fish colliders were scaled down for better performance. Ambiguous gaze directions causing errors were adjusted by repositioning objects. Testing led to gradual introduction of game elements and simplified controls to avoid overloading users.

Two children participated, one with better motor skills and one with motor skills impairment. They were given three training sessions on each game as an intervention. Game performance and accuracy were measured over the sessions. The child with better motor skills adapted to the Leap Motion faster but got distracted easier in later games. The child with motor skills impairment took longer to adapt initially but improved in filtering distractions subsequently. Both subjects showed increased accuracy and speed over the training sessions as they became more proficient.

In conclusion, the motion-based interaction provided an accessible method suitable even for those with poorer motor abilities. The games exhibited potential in developing skills such as sustained focus, multi-tasking, and social communication commonly lacking in ASD. The performance data enabled quantitative progress tracking. Additional research with more varied tasks and subjects would be beneficial to validate clinical viability. Integration of story lines could improve the engagement. With further refinement, such systems could support automated at-home therapy, early screening, and progress monitoring, complementing conventional treatments.

2.2.4.1 Strengths

The use of VR and motion tracking provides an immersive and engaging platform for autism interventions, promoting active participation and motor skill development through practice in a safe simulated environment. The games target training specific real-world skills like sustained focus, filtering distractions, following directions, and recognizing social cues that are often challenging for children with ASD.

2.2.4.2 Weakness

The test results are insufficient to make definitive conclusions on therapeutic efficacy due to small sample size. The games employed simple gameplay, lacking more sophisticated and varied scenarios or tasks to address a broader range of shortcomings. The accuracy of finger motion tracking was limited, reducing fine manipulation capabilities. The dependence on hand controllers reduces accessibility for those with poorer motor control.

2.2.4.3 Recommendation

Broader trials with larger sample sizes of diverse ASD populations are needed to validate efficacy. Gameplay can be enhanced with more dynamic scenarios, customization, and multi-sensory feedback to engage users in training a wider range of real-world skills. Advanced finger tracking and haptics would enable finer motion control and tactile realism. Supporting alternate inputs like eye gaze and adaptive switches would improve accessibility for those with limited motor abilities. Individualized game adaptations tailored to target each user's specific deficits could optimize therapeutic outcomes.

2.2.5 Summary of the Existing Systems

Reviews	Advantages	Disadvantages	Recommendations
Chapter 2.2.1	• High degree of fidelity and interaction for virtual scenarios. • Motion tracking and projection on all sides intensifies the sense of immersion. • Quantitative gains in social, emotional, and adaptation skills.	• High demand of dedicated space. • High implementation and maintenance costs. • Low accessibility • Limited long-term retention & transferability outside VR. • Continuous therapist guidance increases personnel costs. • Scheduling constraints limit family participation.	• Transition training to portable and affordable head-mounted VR systems. • Explore low-cost mobile VR solutions while maintaining immersion levels. • Enhance VR scenarios with varied situations and realistic stimuli. • Implement booster training sessions to sustain gains from training over time. • Supplement therapist support with interactive virtual assistants. • Consider offering VR training outside of school hours.
Chapter 2.2.2	• Real-time integration of facial and voice analytics for emotional assessment.	• Virtual scenarios may not simulate to real-world situations.	• Enhance realism with interactive avatars and environments in virtual scenarios.

	• Provides therapists with quantitative data for measuring progress over time. • VR sessions presented as enjoyable games, fostering active engagement from children. • Enables interactive therapy in remote areas, overcoming geographical constraints.	• Facial and speech recognition is not robust enough. • Oculus Quest 2 headset limited by battery life, potentially restricting session duration. • High cost associated with VR hardware and software requirements.	• Improve emotion and speech recognition using AI trained on child facial and vocal data. • Remove battery limitations by using corded headset alternatives. • Explore more affordable headset and software options. • Optimize VR software for cost-effectiveness and consider financial support strategies like subsidies.
Chapter 2.2.3	• Uses affordable VR hardware. • Readable and recognizable virtual environment for users on the spectrum. • Straightforward setup process. • Practical learning	• Initial anxiety and difficulty with multi-step instructions. • VR content needs modifications for LFA challenges. • Physical discomfort from headset.	• Gradual VR introduction with calming cues. • Break down multi-step instructions into smaller segments. • Provide additional comprehension time and visual aids for directions. • Simplify environments, offer decision guidance, emphasize cause-effect relationships.

	through real-world shopping application.	• Visual occlusion due to interface design.	• Integrate adjustable headset straps, breathable materials, speed toggling, and breaks for comfort. • Decrease clutter, reposition essential objects, use adaptive eye tracking views. • Utilize immersive 3D audio to minimize occlusion.
Chapter 2.2.4	• Immersive VR experience engages users effectively. • Supports motor skill development in a simulated environment. • Targets real-world skills.	• Limited conclusive results due to small sample size. • Games feature basic gameplay and lack diverse scenarios. • Finger motion tracking accuracy is restricted, affecting manipulation. • Dependency on hand controllers limits accessibility.	• Conduct broader trials with larger and diverse ASD populations. • Enhance gameplay with dynamic scenarios and customization. • Incorporate multi-sensory feedback for increased engagement. • Upgrade finger tracking and haptic technology for finer control. • Support alternate inputs for accessibility. • Personalize game adaptations to address individual deficits.

Table 2.2.5.1 Summary of the Existing Systems

2.3 Concluding Remark

The chapter reviews the VR headsets and VR software engines that are exists in the current industry. Then, the chapter also reviews several existing systems that are similar to this project. In this chapter, it also includes the comparison between the reviewed technologies and existing systems.

CHAPTER 3

System Methodology

3.1 System Development Models

3.1.1 Waterfall Model

One of the oldest and most popular approaches to software development is the waterfall model. It is based on the concept of breaking down a project into sequential and separate stages, each with its own set of tasks and deliverables. The waterfall model has a logical and linear flow in which each phase is dependent on the completion and verification of the prior one. The main phases of the waterfall model are requirement analysis, design, implementation, verification, and maintenance [12].

In requirement analysis phase, includes gathering and identifying the software system's functional and non-functional requirements from stakeholders. The requirements are recorded and utilized as the foundation for the software's design and development.

During the design phase, a thorough design document that describes the software architecture, user interface, system components, and integration points is created. The design document acts as a template for the software's development and testing.

Next, the software is coded according to the design specifications during the implementation phase. Unit testing is also performed on the code to ensure that each component functions as intended.

After that, the verification phase will evaluate the program as a whole to ensure that it fits the needs and expectations of the stakeholders. Various testing methods, including acceptance, system, and integration testing, may be used during this phase.

The last phase is the maintenance phase, which involves addressing any errors or flaws detected after the software has been launched. The maintenance phase may also include modifying or improving the program to satisfy new or evolving needs.

The waterfall model works best for software projects with well-defined scope and objectives, as well as predictable outcomes. It is frequently utilized for large-scale, complicated, and safety-critical systems, such as those employed in aerospace, defense, and government projects. However, for software projects that are dynamic, unpredictable, or require frequent modifications or iterations, such as web apps, mobile applications, or agile projects, it may not be the ideal solution.

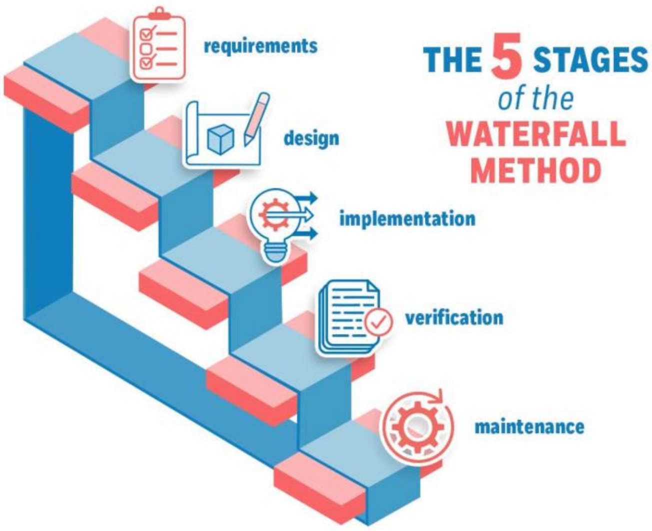


Figure 3.1.1.1 Phases of Waterfall Model

3.1.2 Incremental model

The incremental model is a software development process that is being used to create massive, complicated systems. It is built on the concept of adding new features, or "increments," to an existing system rather than starting from scratch. The incremental model comprises of several phases, including requirement analysis, design & development, testing, and implementation [13].

During the requirement analysis phase, the product needs are determined and separated into several modules that may be built and delivered progressively. Each module is built and executed according to requirements and standards throughout the design and development phase. To verify functionality and compatibility, each module is verified separately during the testing phase and then merged with the current system. Then, each module is deployed to the client location and reviewed for performance and usability during the Implementation phase.

The incremental model is also known as the iterative waterfall model or the successive version model. It is ideal for projects with clear and well-defined criteria, a lengthy development timeframe, a professional and experienced team, and a demand for early value realization.

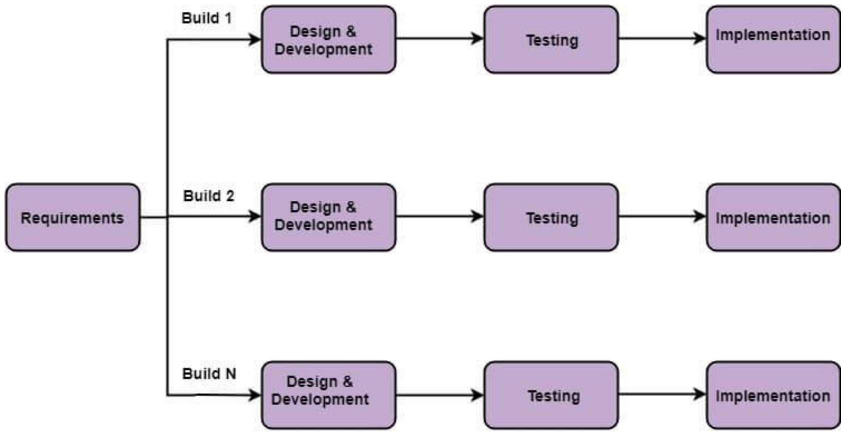


Figure 3.1.2.1 Phases of Incremental Model

3.1.3 Rapid Application Development Model

The goal of rapid application development (RAD), a software development approach, is to generate software products faster and with higher quality than with conventional methods. The ideas of iteration, feedback, and prototyping are key to RAD. These allow developers to quickly build and test software features and functions that meet the needs and expectations of users. RAD's primary phases are requirements planning, user design, construction, and cutover [14].

The project scope, objectives, and requirements are defined and prioritised during the requirements planning process. It also specifies the development process's resources, tools, and methodologies. Next, the team will work with end users to comprehend and confirm their requirements and expectations in user design phase. It makes use of methods including focus groups, brainstorming sessions, workshops, and prototyping. After that, the construction phase involves the creation of the software product based on user design and input. It makes use of cutting-edge development tools and techniques such as visual interface tools, code generators, and reusable components. The software product is tested, deployed, and integrated into the current system during the cutover phase. Additionally, it offers end users documentation, training, and support.

RAD is a suitable framework for software projects with well-defined and stable needs, which can be separated into small and independent modules, that involve client feedback and collaboration throughout the development process, and which have a short time frame and little technical risk. With the aid of RAD, developers may produce software that is more flexible, user-friendly, and sensitive to shifting consumer wants and market trends.

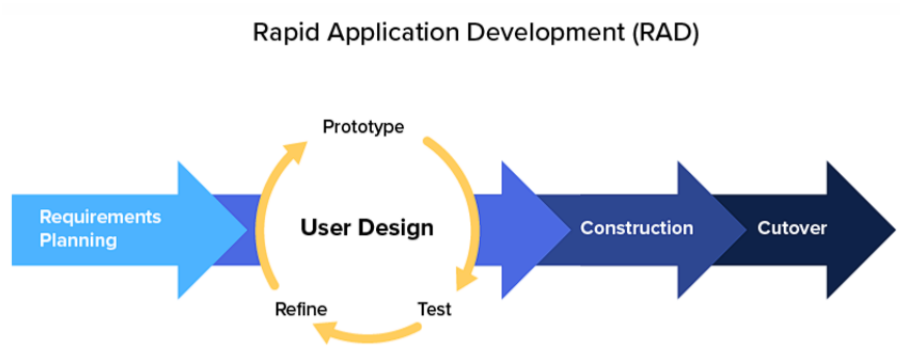


Figure 3.1.3.1 Phases of Rapid Application Development Model

3.1.4 Agile Model

The agile model is a software development methodology that prioritises continuous planning, teamwork, incremental delivery, and ongoing learning. The agile model divides the project into phases, or iterations, which are finished in a short period, often one to four weeks. A team goes through the whole software development cycle at each iteration, which includes requirements gathering, design, construction, testing, deployment, and feedback. After each iteration, the agile model seeks to provide functioning software that fulfills the needs and expectations of the client [15].

During the requirements gathering phase, the needs and expectations of the client are identified and evaluated. Next, the software architecture, parts, and interfaces are specified and prototyped during the design stage. Then, the software code is created, evaluated, and tested throughout the construction process. In the testing phase, the program functionality, performance, usability, security, and dependability are checked and confirmed. After that, the program is delivered and deployed in the target environment during the deployment phase. Customers, users, and stakeholders submit feedback and recommendations for improvement throughout the feedback phase.

Kanban, XP, Crystal, DSDM, and FDD are some popular agile methods. Each method has its own set of practices and roles, but all follow the same agile ideals and principles. The agile model works well for projects with dynamic and changing needs, complex and unpredictable settings, and a high level of client interaction. However, it is not ideal for projects with defined and predictable needs, inflexible and formal processes, and little client engagement.

Figure 3.1.4.1 Phases of Agile Model

3.1.5 Selected Model

The agile model is selected in this project because it is a suitable approach for developing a complex and dynamic VR application that requires frequent feedback and changes. The agile model enables the project to be divided into phases, which allows for the faster delivery of the VR environment. Another reason for employing the agile model is that it delivers software in small phases, reducing risks and uncertainties. Changes can be simply and rapidly implemented in the middle of the process, which allows for flexibility and adaptability. This aids in the correction of any errors that may occur and increases the probability of project success. Furthermore, the agile model also allows for ongoing product improvement and refinement throughout the development process. Any additional corrections and enhancements can be made at any point before the project is launched. This increases the software's quality and functionality.

3.2 System Requirement

3.2.1 Hardware

This project utilizes the hardware components such as laptop and Android smartphone for development and system testing of the VR application. The laptop serves as a capable workstation for software development and testing on this project. It is powered by an Intel Core i5-10500H CPU that able to provide the computing power required for VR application development workloads. Its NVIDIA GeForce RTX 3060 Laptop GPU provides fast graphics rendering, which is essential in producing engaging VR environment. The laptop's 16GB DDR4 RAM and 500GB NVMe M.2 SSD ensure seamless multitasking and rapid data access, respectively.

The Android smartphone is vital for the project's mobile VR software testing. This gadget, which is powered by a Dimensity 8100-Max Octa-core CPU, promises a lot of computational capability. Its 12GB LPDDR5 RAM and 256GB storage capacity add to its overall performance. The integration of the Arm Mali-G610 MC6 GPU improves graphics processing capabilities, which is critical for providing an immersive VR experience. The ColorOS 13.1 operating system powers the Android smartphone.

Table 3.2.1.1 Specifications of Laptop

Description	Specifications
Model	MSI GF65 Thin 10UE
Processor	Intel Core i5-10500H
Operating System	Windows 11
Graphic	NVIDIA GeForce RTX 3060 Laptop GPU
Memory	16GB DDR4 RAM
Storage	500GB NVMe M.2 SSD

Table 3.2.1.2 Specifications of Android Smartphone

Description	Specifications
Model	Oppo Reno 8 Pro
Processor	Dimensity 8100-Max Octa-core
Operating System	ColorOS 13.1

GPU	Arm Mali-G610 MC6
Memory	12GB LPDDR5 RAM
Storage	256GB

3.2.2 Software

The essential software tools used to achieve the project objectives are Unity 2020.3.41f1, the Google Cardboard VR framework, C# programming language and the Blender.

Unity 2020.3.41f1 serves as the project's primary development environment. It is a dependable and versatile platform famous for its outstanding capabilities for handling complex 3D visuals, interactive components, and seamless cross-platform deployment. Its robust toolkit enables the construction of visually appealing landscapes and dynamic interactions, resulting in an engaging user experience. Additionally, Unity's cross-platform functionality makes it easier to realize the project's mobile VR application, which supports a variety of Android devices.

The Google Cardboard VR framework plays an important part in the development process. The framework is designed to create immersive, low-cost VR experiences with excellent accessibility. These deliberately developed settings for Android smartphones provide an engaging and cost-effective medium for autistic primary students to practice and enhance their social skills in a controlled and appealing virtual setting.

The C# programming language is the primary programming medium in Unity. It enables the creation of interactive and dynamic elements inside the VR application. Developers may create complex behaviours, user interfaces, and logical operations that are essential to the project's interaction because of its seamless integration with Unity. This collaboration of Unity, Google Cardboard VR, and C# builds the groundwork for the project's immersive and customizable VR experience. Furthermore, C# code is typically developed and run within Microsoft Visual Studio, which provides a robust environment for debugging and deploying Unity applications.

This project makes significant use of Blender, an open-source 3D computer graphics software, to create the avatar models and animations. Its complete set of modelling capabilities, including polygonal modelling, subdivision surfaces, and sculpting, enabled the creation of highly detailed and visually appealing 3D character models. Blender's node-based material system allowed for broad texturing and shading of avatars to get the desired appearance and feel. The software's powerful animation features helped bring the avatars to life. The rigging and skinning features allowed users to create character rigs using bone structures and map mesh vertices to bones for lifelike deformations during animation. Blender's keyframing technology and graph editor provided exact control over timing and softening of character movements.

3.3 Functional Requirement

3.3.1 Use Case Diagram and Description

In the Figure 3.3.1.1 shows that the user can be either a child or a parent. The parents can select parents then they may modify the classroom, update the child's name, configure sound setting, or change language. If modify classroom option is selected, the parents may change the colour of classroom background, change the number of students' tables or change the colour of the student's table and the teacher's table. In addition, the language of the social story can be switched between English, Chinese, or Malay. The child may choose a social story then it will launch the selected social story. Moreover, the user also has the option to quit the VR application.

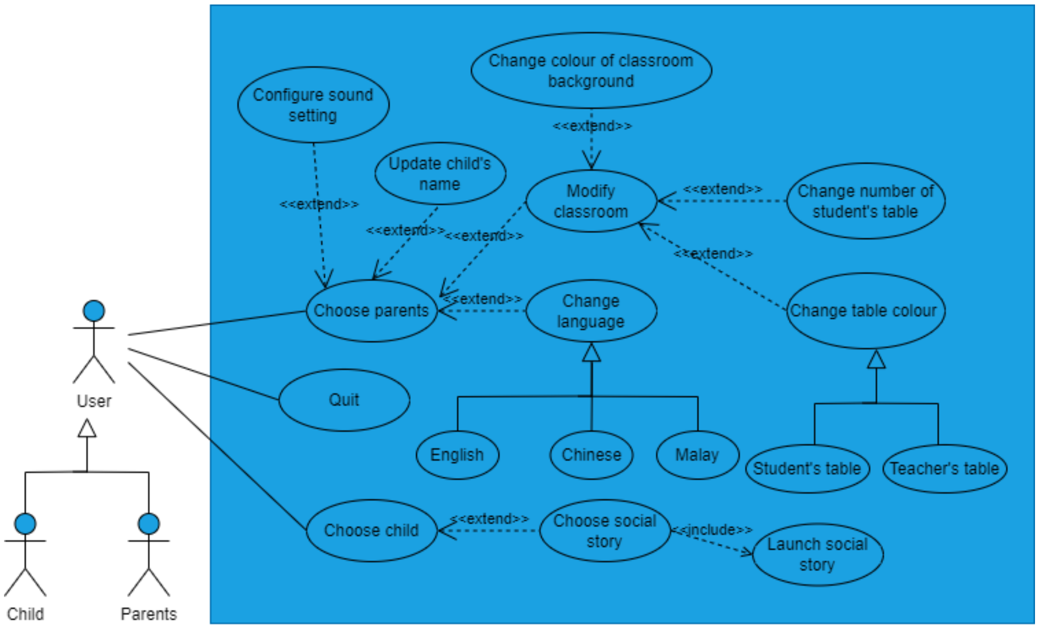


Figure 3.3.1.1 Use Case Diagram

Main Menu

Use Case ID	UC001		Version	1.0
Use Case	Main Menu			
Purpose	To allow the user navigates to the features of the application.			
Actor	User			
Trigger	User successfully launches the application and presents in Main Menu.			
Precondition	User launches the application.			
Scenario Name	**Step**	**Action**		
Main Flow	1	Application displays the options to the user.		
	2	Application prompts the user to select the option.		
	3	Users select an option.		
	4	Application redirects the user to the corresponding feature based on the selected option.		
Sub Flow – Choose Parents	4a.1	User selects the “Choose Parents” option.		
	4a.2	Application redirects user to “Choose Parents” feature.		
Sub Flow – Return Choose Child	4b.1	User selects the “Choose Child” option.		
	4b.2	Application redirects user to “Choose Child” feature.		
Sub Flow Flow – Quit	4c.1	User clicks on exit button.		
	4c.2	User exits the application.		
Rules		-		
Author		Tee Zi Jun		

Table 3.3.1.1 Use Case Description for Main Menu

Choose Parents

Use Case ID	UC002	Version	1.0
Use Case	Choose Parents		
Purpose	To allow the user navigates to the customizable features of the application.		
Actor	User		
Trigger	User selects the “Choose Parents” option in Main Menu.		
Precondition	User launched the application and currently is in the Main Menu.		

Scenario Name	Step	Action
Main Flow	1	Application displays the options to the user.
	2	Application prompts the user to select the option.
	3	Users select an option.
	4	Application redirects the user to the corresponding feature based on the selected option.
Sub Flow – Sound Setting	4a.1	User selects the “Sound Setting” option.
	4a.2	Application redirects user to “Sound Setting” feature.
	4a.3	In sound setting, user can use slider to adjust the volume of background music.
	4a.4	After configuration, user clicks on back button.
	4a.5	Application redirects user back to “Choose Parents” feature.
Sub Flow – Update Child’s Name	4b.1	User selects the “Update Child’s Name” option.
	4b.2	Application redirects user to “Update Child’s Name” feature.
	4b.3	User can type their child’s name in the text field.
	4b.4	User clicks on “Save” button.
	4b.5	Application updates the child’s name into social story.
	4b.6	User clicks on back button.
	4b.7	Application redirects user back to “Choose Parents” feature.
Sub Flow – Classroom	4c.1	User selects the “Classroom” option.
	4c.2	Application redirects user to “Classroom” feature.
	4c.3.i	User changes the colour of classroom background using sliders.
	4c.3.ii	User changes the number of student’s table.
	4c.3.iii	User changes the colour of student’s table.
	4c.3.iv	User changes the colour of teacher’s table.
	4c.4	After customization, user clicks on back button.
	4c.5	Application redirects user back to “Choose Parents” feature.

Sub Flow – Language Setting	4d.1	User selects the "Language Setting" option.
	4d.2	Application redirects user to "Language Setting" feature.
	4d.3.i	User selects English as the language of the social story.
	4d.3.ii	User selects Chinses as the language of the social story.
	4d.3.iii	User selects Malay as the language of the social story.
	4d.4	After language is selected, user clicks on back button.
	4d.5	Application redirects user back to "Choose Parents" feature.
Sub Flow – Back to Main Menu	4e.1	User clicks on back button.
	4e.2	Application redirects user back to Main Menu.
Rules		-
Author		Tee Zi Jun

Table 3.3.1.2 Use Case Description for Choose Parents

Choose Child

Use Case ID	UC003	**Version**	1.0
Use Case	Choose Child		
Purpose	To allow the user navigates to the selection of social stories.		
Actor	User		
Trigger	User selects the "Choose Child" option in Main Menu.		
Precondition	User launched the application and currently is in the Main Menu.		
Scenario Name	**Step**	**Action**	
Main Flow	1	Application displays the options to the user.	
	2	Application prompts the user to select the option.	
	3	Users select an option.	
	4	Application redirects the user to the corresponding feature based on the selected option.	
Sub Flow – Choose Social Story	4a.1	User chooses the social story.	
	4a.2	Application launches the selected social story.	
	4a.3	User clicks on back button.	
	4a.4	Application redirects user to "Choose Child" feature.	

Sub Flow – Back to Main Menu	4b.1	User clicks on back button.
	4b.2	Application redirects user back to Main Menu.
Rules		-
Author		Tee Zi Jun

Table 3.3.1.3 Use Case Description for Choose Child

3.3.2 Activity Diagram

Main Menu

Based on Figure 3.3.2.1, it shows the activity diagram for Main Menu starts with user launches the VR application. The VR application displays the options to user then prompts the user to select an option. If user selects the “Choose Parents” option, application will redirect the user to “Choose Parents” feature. If user selects the “Choose Child” option, application will redirect the user to “Choose Child” feature. If user clicks on exit button, user will exit the application.

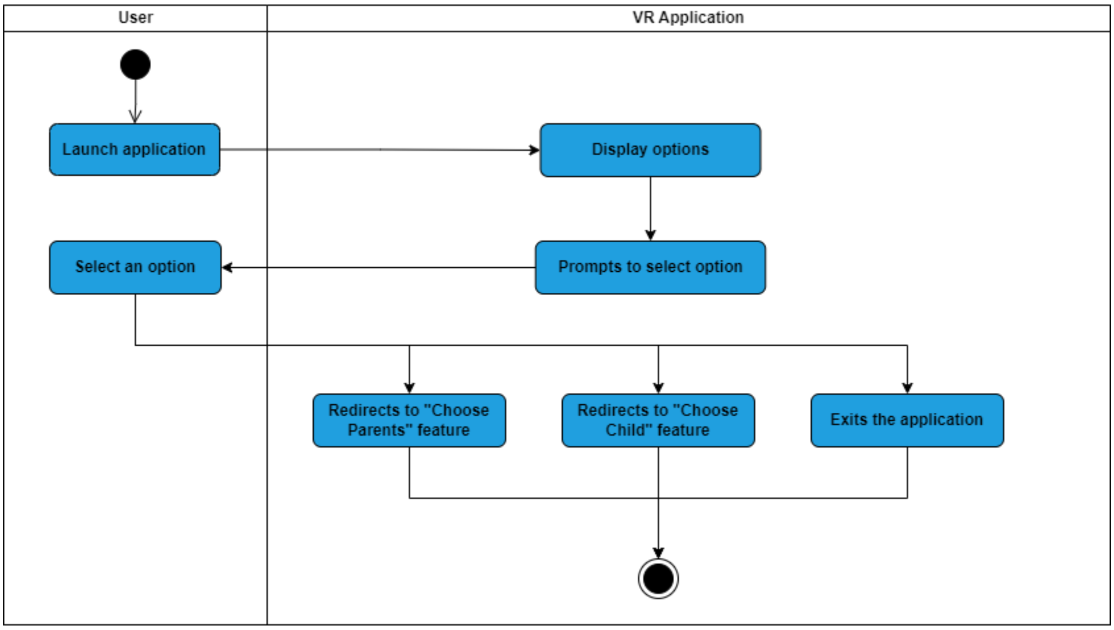


Figure 3.3.2.1 Activity Diagram for Main Menu

Choose Parents

Figure 3.3.2.2 shows the activity diagram for Choose Parents begins with VR application displays the options then prompts the user to select options. If user selects “Sound Setting” option, application will redirect user to “Sound Setting” feature. User can adjust the volume of background music within sound setting. If user selects “Update Child’s Name” option, application will redirect user to “Update Child’s Name” feature. User can type their child’s name in text field then click on “Save” button to update their child’s name into social story. If user selects “Classroom” option, application will redirect user to “Classroom” feature. User can change the classroom background colour, choose the number of student’s table, change the colour of student’s table or change the colour of teacher’s table. If user selects “Language Setting” option, application will redirect user to “Language Setting” feature. User can choose the English, Chinese or Malay as the language of the social story. After the configuration in “Sound Setting” feature, “Update’s Child Name” feature, “Classroom” feature or “Language Setting, user can click on back button then application will redirect user back to “Choose Parents” feature. If user clicks on back button in “Choose Parents” feature, application will redirect user to Main Menu.

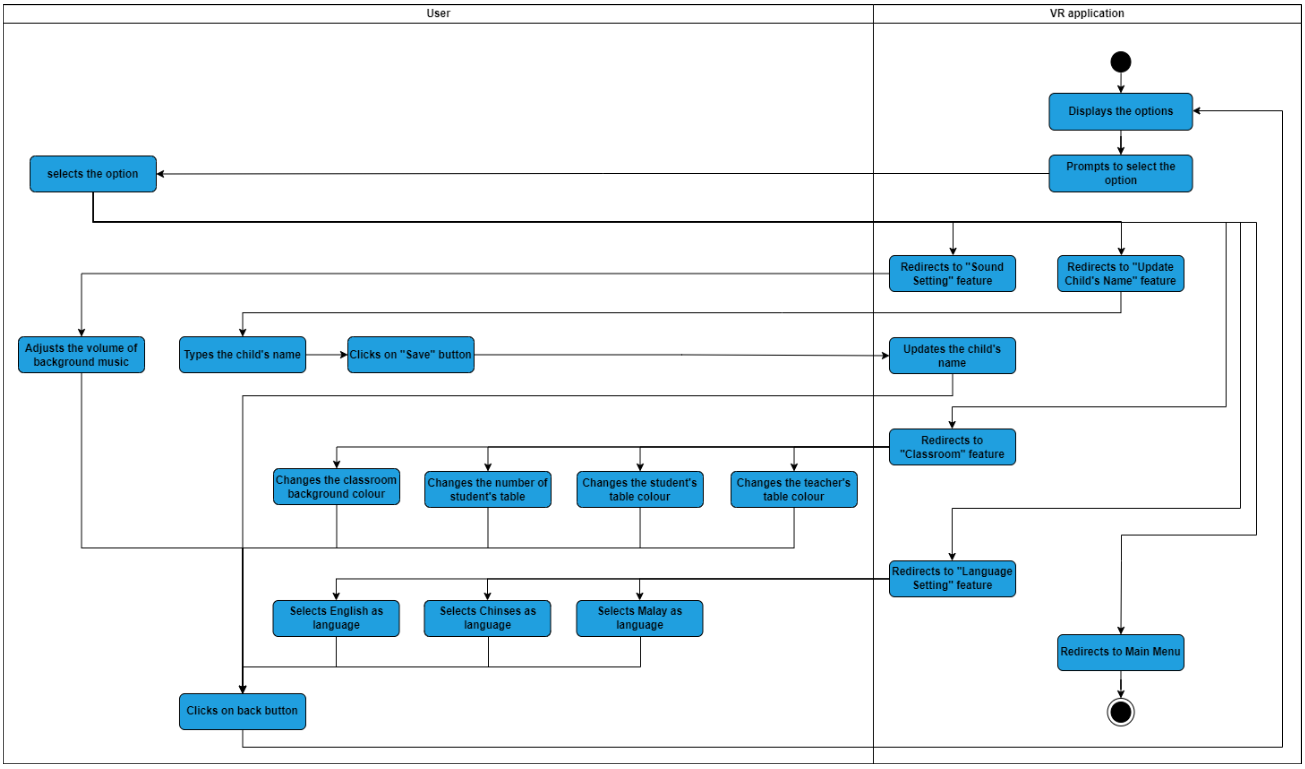


Figure 3.3.2.2 Activity Diagram for Choose Parents

Choose Child

In Figure 3.3.2.3, the activity diagram for Choose Child starts with VR application displays the options then prompts the user to select options. If user selects one of the social stories, VR application will launch the selected social story. After selected social story ended, user clicks on back button to redirect back to "Choose Child" feature. If user clicks on back button in "Choose Child" feature, VR application will redirect user to Main Menu.

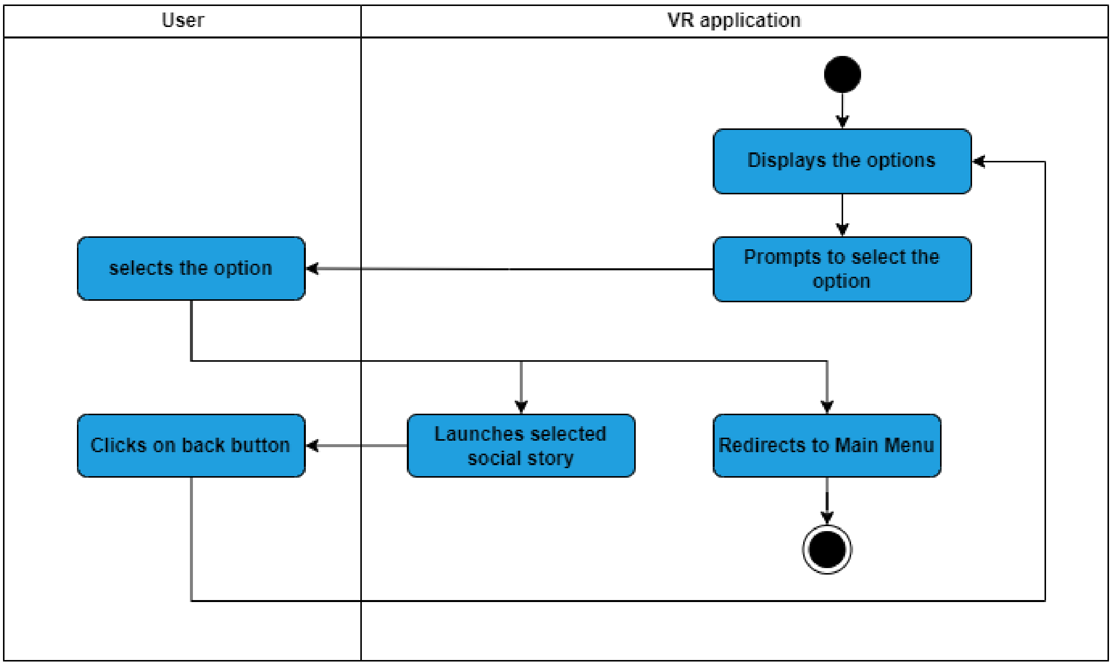


Figure 3.3.2.3 Activity Diagram for Choose Child

3.4 Expected System Testing and Performance

Table 3.2.1 will show the test cases for the VR application to check whether the system meets the functional requirements or specifications. System testing would assist in finding and resolving any flaws or mistakes in the system, as well as enhancing its quality and usefulness.

No	Test Cases	Expected Result
1	Click on the “Parents” option in main menu.	Direct to environment menu and display options such as classroom, language, and update child’s name.
2	Click on the “Classroom” option in environment menu.	Direct to classroom setting menu.
3	Click on the “Classroom Background” option in classroom setting menu.	Direct to classroom background setting interface.
4	Change the colour of classroom background using the sliders in classroom background setting interface.	The colour of classroom background in social story should be updated.
5	Click on back button in classroom background setting interface.	Direct to the classroom setting menu.
6	Click on the “Objects in the Classroom” option in classroom setting menu.	Direct to modify classroom objects interface.
7	Choose the number of student’s table in modify classroom objects interface.	Spawn the student’s table according to the selected number.
8	Choose the colour of student’s table in modify classroom objects interface.	The student’s table change to selected colour.
9	Choose the colour of teacher’s table in modify classroom objects interface.	The teacher’s table change to selected colour.

10	Click on back button in modify classroom objects interface.	Direct to the classroom setting menu.
11	Click on back button in the classroom setting menu.	Direct to the environment menu.
12	Click on the "Language" option in environment menu.	Direct to the language menu.
13	Choose any of the languages (English, Chinese, Malay) in the language menu.	The language of social story should change to the selected language.
14	Click on back button in language menu.	Direct to the environment menu.
15	Click on the "Sound Setting" option in environment menu.	Direct to the sound setting menu.
16	Adjust the background music volume using slider in sound setting menu.	The volume of background music in social story should be updated.
17	Click on back button in sound setting menu.	Direct to the environment menu.
18	Click on update child's name in environment menu.	Direct to the update child's name menu.
19	Enter the child's name in the input field then hit enter.	The social story should mention the updated child's name.
20	Click on back button in update child's name menu.	Direct to the environment menu.
21	Click on back button in environment menu.	Direct to the main menu.
22	Click on the "Child " option in main menu.	Direct to story selection menu.
23	Click on any social story option in story selection menu.	It will launch the selected social story.
24	Click on back button after the social story is finished	Direct to story selection menu.

25	Click on back button in story selection menu.	Direct to the main menu.
26	Click on the “x” button in main menu.	Quite the application.

Table 3.4.1 Test Cases for VR Application

The expected performance is used to assess how effectively the system can handle various scenarios and circumstances that may arise when users interact with the VR world. Here are some of the expected performances:

- Without any delays or problems, the system should be able to respond quickly to the user's requests or actions, such as changing the child's name or changing the classroom.
- The system should be able to consume minimal resources when running, such as CPU, memory, disk, network, and so on, to guarantee that the system is efficient and stable.
- The system should be able to handle several languages and show text as well as social stories in the relevant and suitable language.
- The system should be able to render the virtual environment, avatars, and objects smoothly, without any noticeable lag or stuttering.
- The system should be compatible with a wide range of devices and display resolutions, ensuring a consistent experience for users with different hardware configurations.
- The system should provide a seamless and pleasant experience for users, as well as be capable of handling customisation choices and several languages without problems or malfunctions.

3.5 Expected Challenges

The first challenge faced in the development process is choosing the right tone, tempo, and delivery style for the voiceovers is critical to properly engaging autistic primary school students. Finding the proper balance and ensuring that the voiceovers resonate with the intended audience may be a challenging task without prior expertise. Poor audio quality can be a significant distraction, reducing the overall efficacy of social stories. Furthermore, integrating the audio voiceovers seamlessly with the visuals and timing of the social stories requires precise coordination and attention to detail. Misalignment or incorrect synchronization of audio and visual elements can interrupt the flow and comprehension of stories, reducing their impact. The difficulty is further complicated by the need to account for possible language variations in the voiceovers for various localization settings.

Another challenge in this project introducing interactive elements into social stories while keeping the storyboard style. The fundamental goal of the storyboard style is to allow autistic primary school students to concentrate on the social story without distractions. However, integrating interactive elements to increase engagement and skill development may contradict with this goal, requiring a fine balance to be struck. Determining the right amount and type of interactions that can improve learning without affecting the overall experience is essential. The interactive elements must be adjusted to the target audience's specific requirements and skills, ensuring that it can integrate smoothly into the entire experience without generating obstacles or misunderstanding.

The last challenge that expected to face in this project is a comprehensive understanding of the thought processes and emotional triggers of autistic students is necessary when crafting or modifying social stories to meet their specific learning demands. It can be difficult to find the ideal balance between educational value and simplicity, particularly when considering the range of demands associated with ASD. Some primary students with ASD, for example, may require social stories that emphasize the positive outcomes and rewards of a social context, whereas others may want social stories that emphasize the potential risks and impacts of a social situation. It can be tough and laborious to create or adjust the social story without taking these things into account.

3.6 Project Milestone

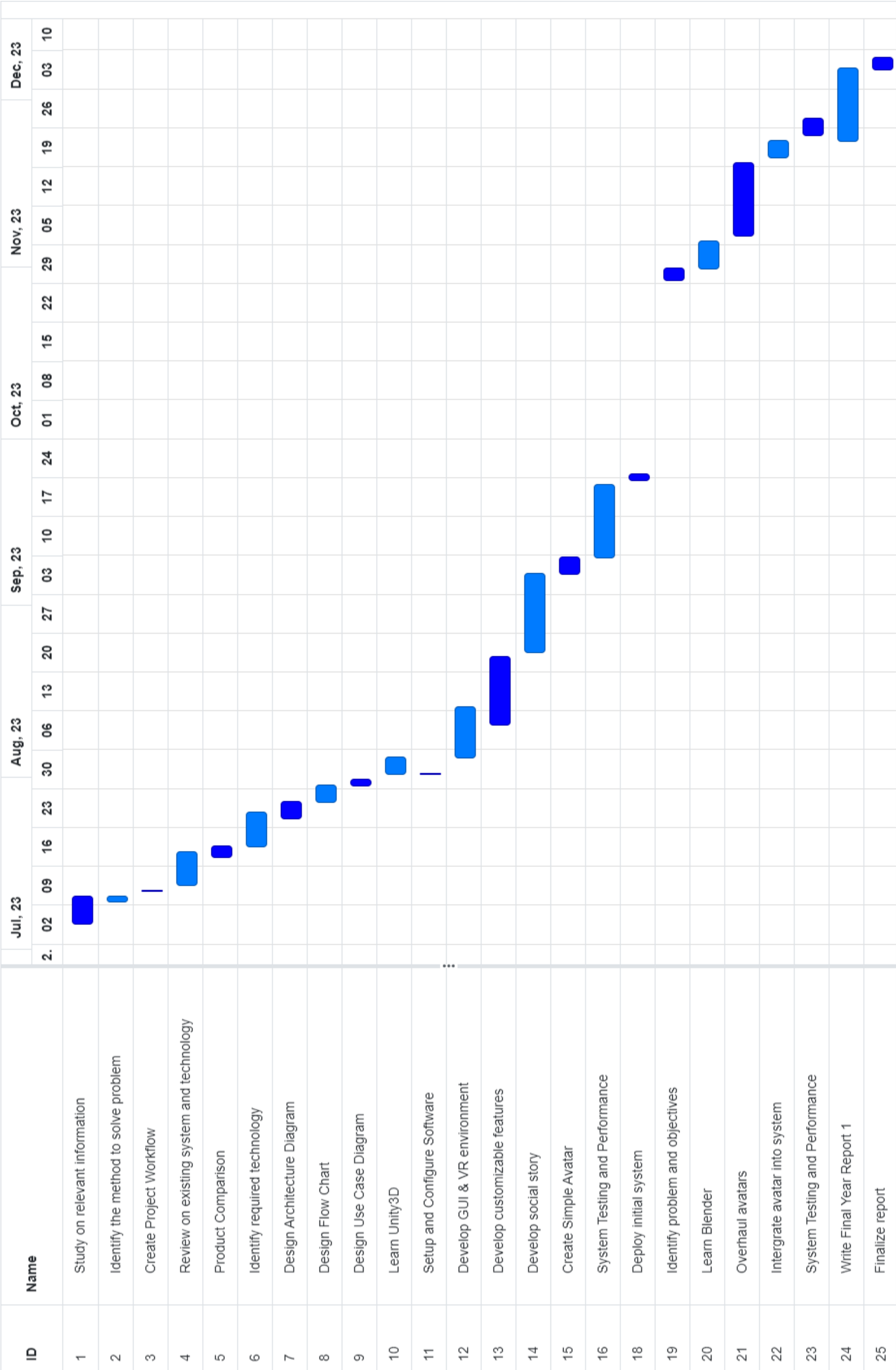


Figure 3.6.1 Gantt Chart 1

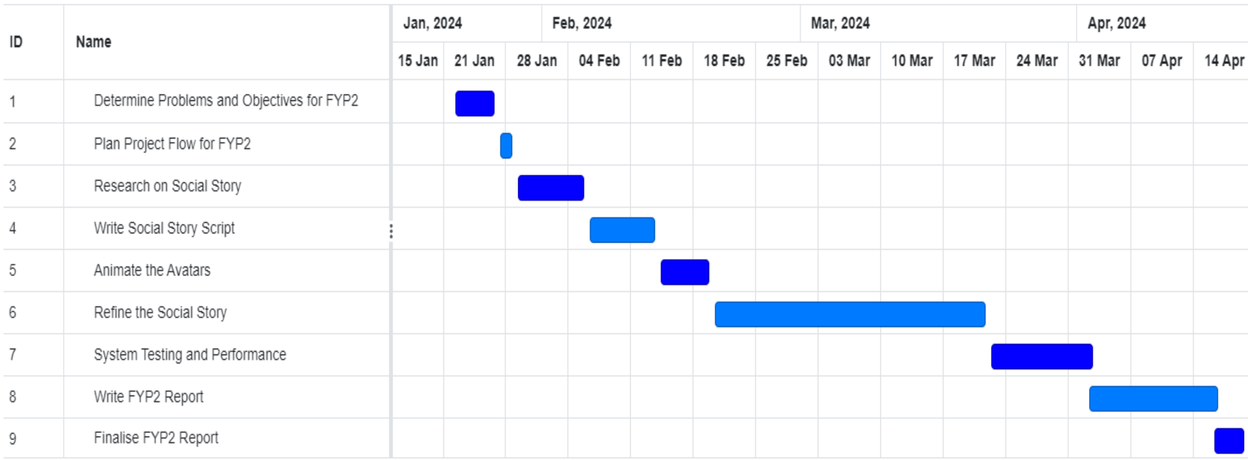


Figure 3.6.2 Gantt Chart 2

3.7 Concluding Remark

This chapter will review the existing system development models and select the suitable system development model for this project. Additionally, the system requirements such as hardware and software will be shown in this chapter. The functional requirements are expressed in the form of a use case diagram and its description as well as activity diagrams. This chapter also includes the expected system testing, performance, and challenges along with the project milestone.

CHAPTER 4

System Design

4.1 System Architecture

In Figure 4.1.1, the VR application, the device, the database, and the VR platform are the four primary parts of the VR project. The VR application is the software that powers the VR experience and delivers the UI, 3D visuals, audio, interactivity, and logic. The device is the hardware that the user wears or uses to access the VR experience, such as a headset, phone, or computer. The database service is responsible for storing and retrieving data connected to the VR experience, such as VR content and user settings. The VR platform is the software that offers common functions and services for the VR application, such as the scene graph, rendering engine, physics engine, and network connection.

The VR application interfaces with the device via the VR platform, which isolates the low-level features of the hardware and offers a high-level interface for the VR application. The VR application may both access and upload data to the database. The VR platform may also interface with the database, for example, to download new VR content or to update the VR application.

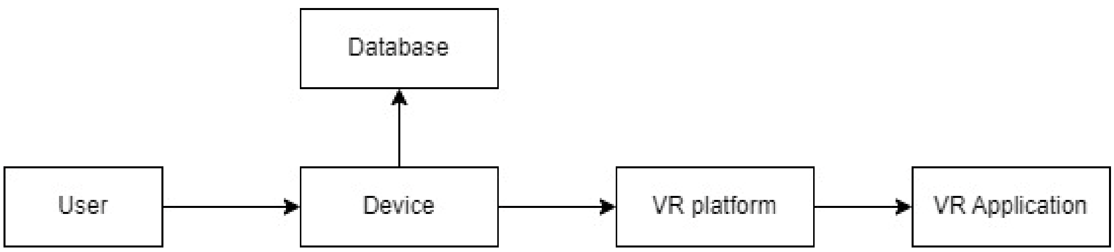


Figure 4.1.1 Block Diagram

4.2 Functional Modules in the System

Functional Module	Description
Background lighting	This module manages the VR environment's light sources' hue, brightness, and direction. It may also modify the lighting based on the context of the social story.
Avatar	This module generates and animates 3D models of the user and other characters in the virtual reality environment.
Audio	This module offers the VR experience with audio voiceovers and background music. It can generate realistic audio voiceovers and interesting background music that are appropriate for the VR experience. The volume, pitch, and direction of the noises may also be adjusted according on the user's location and orientation.
Locomotion	This module allows the user to navigate the VR world. Depending on the setting of the social story, it may provide various ways of mobility like as teleportation or walking.
Data storage	This module saves and retrieves information about the VR experience. It has the ability to save the user's settings or VR content in the database service. It can also access the database service's data and upload new data to the database service.
Social story	This module provides the story and interaction for the VR experience. It can choose a social story based on the parent's preferences. It can also assist the user through the social story using voice, text, or visual clues.
User interface	This module contains the graphical elements and menus for the VR experience. It can display information such as social narrative options and VR settings. It can also allow the user to interact with the VR application, such as selecting options, changing settings, or exiting the VR experience.

Table 4.2.1 Functional Modules in VR Environment

4.3 System Flow

Based on Figure 4.3, user will enter main menu after running the VR application. If the user is parents, the "Parents" option can be chosen to modify the classroom environment, change language, update child's name, and adjust sound setting. If the user is a child, the "Child" option is selected then the child can choose the social stories. After the social story is chosen, it will launch the social story immediately.

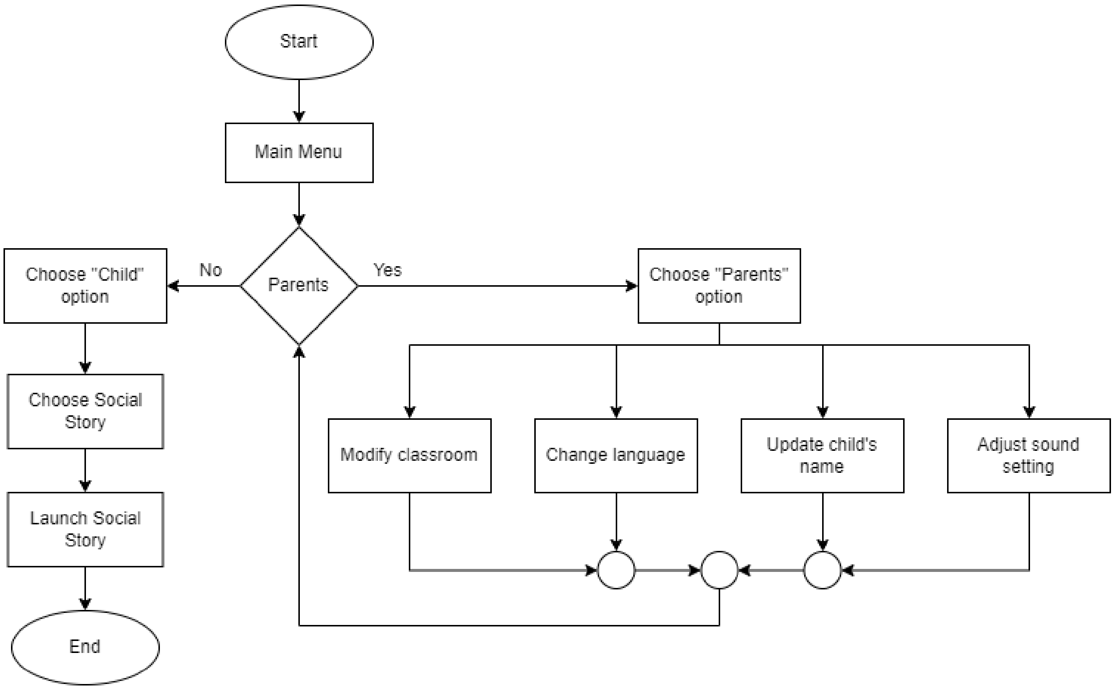


Figure 4.3.1 Flowchart

4.4 Algorithm Design

Below is the pseudocode for the C# scripts that interact with the elements in Unity:

// Import necessary libraries such as System, UnityEngine,etc

// Define a public class for the script that inherits from MonoBehaviour

public class {Class Name} : MonoBehaviour

{

//For switch scene script

```
function GoToScene(sceneName)
   SceneManager.LoadScene(sceneName) // Load the specified scene

function GoToLoadingScene(storyNumber)
   PlayerPrefs.SetInt("StoryChosen", storyNumber) // Set the chosen story
number in player preferences
   SceneManager.LoadScene("LoadingScene") // Load the loading scene

function QuitGame()
   Application.Quit() // Quit the application
```

//For loading scene script

```
function Start()
   StartCoroutine(LoadSceneAfterDelay()) // Start the coroutine to load the
scene after a delay

function LoadSceneAfterDelay()
   storyNum = PlayerPrefs.GetInt("StoryChosen") // Get the chosen story
number from player preferences
   yield return WaitForSeconds(5f) // Wait for 5 seconds
   // Determine which scene to load based on the story number
   switch (storyNum)
      case (storyNum):
         // Load the corresponding story scene
         SceneManager.LoadScene("{StorySceneName}")
         break
```

//For next story scene script

```
function Start()
   // Get the name of the current active scene
   currentSceneName = SceneManager.GetActiveScene().name
   // Determine the next scene and the delay based on the current scene
   switch (currentSceneName)
      case "{ currentSceneName}":
         secondWait = {seconds} // based on duration of scene
         StartCoroutine(DelayFunction(secondWait))
         nextSceneName = " NextSceneName "
         break

function DelayFunction(seconds)
```

```
    // Wait for the specified number of seconds
    yield return WaitForSeconds(seconds)
    // Load the next scene
    loadNextScene(nextSceneName)

function loadNextScene(sceneName)
    // Load the specified scene using a function from PanelManager
    PanelManager.Instance.LoadNextScene(sceneName)
```

//For exit to previous screen script

```
function Start()
    // Get the index of the current active scene
    sceneIndex = SceneManager.GetActiveScene().buildIndex

function Update()
    // Check if the Escape key is pressed
    if Input.GetKeyDown(KeyCode.Escape)
        // Load the previous scene by subtracting 1 from the current scene index
        SceneManager.LoadScene(sceneIndex - 1)
```

//For classroom scene script

```
function Start()
    // Get the index of the current active scene
    sceneIndex = SceneManager.GetActiveScene().buildIndex
    // Set the classroom background color based on player preferences
    chosenColor = GetChosenColorFromPlayerPrefs()
    SetClassroomBackgroundColor(chosenColor)
    // Load objects in the classroom scene if applicable
    LoadObjectsIfNeeded()

function GetChosenColorFromPlayerPrefs()
    // Retrieve color values from player preferences, default to white if not
found
    r = PlayerPrefs.GetFloat("ColorR", 1)
    g = PlayerPrefs.GetFloat("ColorG", 1)
    b = PlayerPrefs.GetFloat("ColorB", 1)
    return new Color(r, g, b)

function SetClassroomBackgroundColor(color)
    // Set the background color of classroom walls
    wall1.material.color = color
    wall2.material.color = color
    wall3.material.color = color

function LoadObjectsIfNeeded()
    // Check if objects need to be instantiated in the classroom
    objectsInstantiated = PlayerPrefs.GetInt("ObjectsInstantiated", 0)
    teacherTableInstantiated = PlayerPrefs.GetInt("TeacherTableInstantiated",
0)
    // If objects need to be instantiated, load them
```

```
    if objectsInstantiated == 1
        LoadStudentTables()
    if teacherTableInstantiated == 1
        LoadTeacherTable()

function LoadStudentTables()
    // Load student tables and chairs into the classroom
    assetFileName = PlayerPrefs.GetString("myAssetFile")
    valueRow = PlayerPrefs.GetInt("valueRow", 0)
    valueColumn = PlayerPrefs.GetInt("valueColumn", 0)
    // Load asset file and student prefab
    assetFind = LoadAsset(assetFileName)
    studentPrefab = LoadStudentPrefab()
    // Instantiate student tables and chairs based on specified row and column
values
    InstantiateStudentTablesAndChairs(assetFind, studentPrefab, valueRow,
valueColumn)

function LoadAsset(fileName)
    // Load asset file from resources
    return Resources.Load<GameObject>(fileName)

function LoadStudentPrefab()
    // Load student prefab from resources
    return Resources.Load<GameObject>("Student/studentSit")
function InstantiateStudentTablesAndChairs(asset, studentPrefab, row,
column)
    // Calculate positioning parameters
    xOffset = plane.transform.position.x
    yOffset = plane.transform.position.y
    zOffset = plane.transform.position.z
    spacing = 1.5f
    zPosForRow = -4
    // Loop through rows and columns to instantiate tables and chairs
    for rowCounter from 0 to row - 1
        for columnCounter from 0 to column - 1
            // Instantiate student table
            InstantiateTable(asset, xOffset, yOffset, zPosForRow)
            // Instantiate student chair
            InstantiateStudent(studentPrefab, xOffset, yOffset, zPosForRow)
            // Update positioning parameters
            IncrementPositioningParameters()

function InstantiateTable(asset, xOffset, yOffset, zPosForRow)
    // Instantiate student table at specified position
    Instantiate(asset, new Vector3(xOffset, yOffset, zPosForRow),
Quaternion.identity)

function InstantiateStudent(studentPrefab, xOffset, yOffset, zPosForRow)
    // Instantiate student chair at specified position
```

```
    Instantiate(studentPrefab, new Vector3(xOffset, yOffset, zPosForRow +
0.35f), Quaternion.identity)

function IncrementPositioningParameters()
    // Update positioning parameters for next table/chair instantiation
    zPosForRow += 2

function LoadTeacherTable()
    // Load teacher table into the classroom
    teacherTableFile = PlayerPrefs.GetString("teacherTableFile")
    // Load teacher table asset and instantiate it
    assetFind = LoadAsset(teacherTableFile)
    InstantiateTeacherTable(assetFind)

function InstantiateTeacherTable(asset)
    // Instantiate teacher table at specified position and scale
    myObj = Instantiate(asset)
    myObj.transform.localScale *= 1.5f
    myObj.transform.position = new Vector3(5, 2.2f, -8)
    myObj.transform.rotation = Quaternion.Euler(0f, 90f, 0f)
```

//For classroom background script

```
function Awake()
    // Retrieve saved color values from PlayerPrefs and set slider values
accordingly
    r = PlayerPrefs.GetFloat("ColorR")
    g = PlayerPrefs.GetFloat("ColorG")
    b = PlayerPrefs.GetFloat("ColorB")
    if r != null and g != null and b != null
        // Set slider values to saved color values
        red.value = r
        green.value = g
        blue.value = b
    else
        // Set default slider values to white
        red.value = 1
        green.value = 1
        blue.value = 1

function Update()
    // Update the background color using the current slider values
    imageRenderer.material.color = new Color(red.value, green.value,
blue.value)

function UpdateBgColor()
    // Update the background color in PlayerPrefs with the current slider values
    PlayerPrefs.SetFloat("ColorR", red.value)
    PlayerPrefs.SetFloat("ColorG", green.value)
    PlayerPrefs.SetFloat("ColorB", blue.value)
    PlayerPrefs.Save()
```

```
    // Show a message indicating that the color has been updated
    ShowColorUpdateMessage()
```

//For the language script

```
function Awake()
    // Allocate the language file path according to the language selected from
PlayerPrefs
    languageSelected = PlayerPrefs.GetString("Language")
    if languageSelected != null
        if languageSelected == "Chinese"
            languageFilePath = "Language/Chinese"
        else if languageSelected == "English"
            languageFilePath = "Language/English"
        else if languageSelected == "Melayu"
            languageFilePath = "Language/Melayu"
        Debug.Log("languageFilePath: " + languageFilePath)
    else
        // Default to English if no language selected
        languageSelected = "English"
        languageFilePath = "Language/English"

function Start()
    currentSceneName = SceneManager.GetActiveScene().name
    if currentSceneName != "LanguageSelectionScene"
        if languageFilePath != null
            LoadLanguageFile(languageFilePath)
        // Find the canvas
        canvas = FindObjectOfType<Canvas>()
        if canvas != null
            // Load and display text in the story scene only
            LoadAndDisplayText(canvas)

function LoadLanguageFile(filePath)
    // Load and parse the language file
    languageDictionary = new Dictionary<string, string>()
    textAsset = Resources.Load<TextAsset>(filePath)
    if textAsset != null
        fileContent = textAsset.text
        childName = PlayerPrefs.GetString("ChildName")
        if childName != null
            fileContent = fileContent.Replace("[replace name here]", childName)
        else
            fileContent = fileContent.Replace("[replace name here]", "New
Kiddo")
        lines = fileContent.Split('\n')
        foreach line in lines
            keyValue = line.Split('=')
            if keyValue.Length == 2
                key = keyValue[0].Trim()
                value = keyValue[1].Trim()
```

```
            languageDictionary.Add(key, value)

function LoadAndDisplayText(canvas)
    // Load and display text on panels in the canvas
    panel = PanelManager.Instance.GetPanelByName("TextBubble")
    panel2 = PanelManager.Instance.GetPanelByName("TextBubble2")
    panel3 = PanelManager.Instance.GetPanelByName("TextBubble3")
    LoadAndDisplayTextOnPanel(panel)
    LoadAndDisplayTextOnPanel(panel2)
    LoadAndDisplayTextOnPanel(panel3)

function LoadAndDisplayTextOnPanel(panel)
    if panel != null
        panel.gameObject.SetActive(true)
        textMeshProArray =
panel.GetComponentsInChildren<TextMeshProUGUI>(true)
        foreach textMeshPro in textMeshProArray
            textMeshPro.text = GetLocalizedString(textMeshPro.name)
            SetFontBasedOnLanguage(textMeshPro)
            Debug.Log("TextMeshPro found: " + textMeshPro.name + "\nText: "
+ textMeshPro.text)

function GetLocalizedString(key)
    // Retrieve the localized string for the given key from the language
dictionary
    if languageDictionary.ContainsKey(key)
        return languageDictionary[key]
    else
        return "Localization not found for key: " + key

function SetFontBasedOnLanguage(textMeshPro)
    // Set the font of the TextMeshPro based on the selected language
    if languageSelected == "Chinese"
        textMeshPro.font = fontAssetChinese
    else if languageSelected == "English" or languageSelected == "Melayu"
        textMeshPro.font = fontAssetEnglish

function UpdateText()
    // Update the text on canvas (for testing purposes)
    textUpdated = GetLocalizedString("greeting")
    Debug.Log("The text is: " + textUpdated)
    canvas = FindObjectOfType<Canvas>()
    if canvas != null
        textMeshPro = canvas.GetComponentInChildren<TextMeshProUGUI>()
        if textMeshPro != null
            textMeshPro.text = languageSelected

function LanguageSelection()
    // Update language selection based on button clicked
    clickedButton = EventSystem.current.currentSelectedGameObject
```

```
    if clickedButton != null
        button = clickedButton.GetComponent<Button>()
        if button != null
            if button.name == "ChineseButton"
                PlayerPrefs.SetString("Language", "Chinese")
                ShowLanguageUpdateMessage("Chinese")
            else if button.name == "EnglishButton"
                PlayerPrefs.SetString("Language", "English")
                ShowLanguageUpdateMessage("English")
            else if button.name == "MelayuButton"
                PlayerPrefs.SetString("Language", "Melayu")
                ShowLanguageUpdateMessage("Melayu")

function ShowLanguageUpdateMessage(language)
    // Show a message indicating the language update
    Debug.Log("Language update to " + language)
    SSTools.ShowMessage("Language Updated To " + language,
SSTools.Position.bottom, SSTools.Time.oneSecond)
```

//For modify classroom object script

```
function Start()
    // Load sprites for student tables and teacher tables dynamically
    LoadStudentTableSprites()
    LoadTeacherTableSprites()
    // Generate buttons for student tables and teacher tables
    CreateStudentTableButtons()
    CreateTeacherTableButtons()
    // Check if objects have already been instantiated
    if ObjectsNotInstantiated()
        // If not, instantiate objects based on user preferences
        InstantiateObjects()

function LoadStudentTableSprites()
    // Load sprites (student tables) dynamically at runtime
    for each button sprite in buttonSprites array
        Load and assign sprite to buttonSprites array element

function LoadTeacherTableSprites()
    // Load sprites (teacher tables) dynamically at runtime
    for each button sprite in buttonSprites2 array
        Load and assign sprite to buttonSprites2 array element

function CreateStudentTableButtons()
    // Create buttons for student tables
    for i from 1 to 6
        CreateButtonForStudentTable(i)

function CreateButtonForStudentTable(index)
    // Instantiate a button for student table
    button = Instantiate buttonPrefab1
```

```
    Set button's name
    Set button's image sprite using buttonSprites array
    Set button's position in the panel
    Add listener for button's onClick event to ButtonClicked function with
index

function CreateTeacherTableButtons()
    // Create buttons for teacher tables
    for i from 1 to 3
        CreateButtonForTeacherTable(i)

function CreateButtonForTeacherTable(index)
    // Instantiate a button for teacher table
    button = Instantiate buttonPrefab2
    Set button's name
    Set button's image sprite using buttonSprites2 array
    Set button's position in the panel
    Add listener for button's onClick event to DisplayTeacherTable function
with index

function ButtonClicked(buttonIndex)
    // Handle button click event for student tables
    assetFileName = "Table/desk" + buttonIndex
    Find and load GameObject based on assetFileName
    Store assetFileName in PlayerPrefs
    Call LoopToDisplay function with loaded GameObject

function LoopToDisplay(assetFind)
    // Instantiate student tables based on user preferences
    Calculate positions and instantiate tables according to rows and columns
    Update PlayerPrefs with instantiated objects information

function DisplayTeacherTable(index)
    // Handle button click event for teacher tables
    assetFileName = "TeacherTable/" + index
    Find and load GameObject based on assetFileName
    Instantiate and position teacher table
    Update PlayerPrefs to indicate teacher table instantiation

function DropdownValueUpdateRow(index)
    // Handle dropdown value change event for rows
    Update valueRow based on selected index
    Store valueRow in PlayerPrefs

function DropdownValueUpdateColumn(index)
    // Handle dropdown value change event for columns
    Update valueColumn based on selected index
    Store valueColumn in PlayerPrefs

function ObjectsNotInstantiated()
```

```
    // Check if objects have already been instantiated
    if PlayerPrefs does not contain "ObjectsInstantiated"
        Set PlayerPrefs "ObjectsInstantiated" to 1
        return true
    else
        return false
```

//For update child’s name script

```
function Start()
    // Set character limit for child name input field
    SetCharacterLimitForChildNameField()

function saveChildNameButtonClicked()
    // Handle save button click event
    Retrieve child name from input field
    if child name is not empty
        Save child name in PlayerPrefs
        Show message indicating child's name update
    else
        Save default child name as "New Kid" in PlayerPrefs

function clearButtonClicked()
    // Handle clear button click event
    Clear text in child name input field

function SetCharacterLimitForChildNameField()
    // Set character limit for child name input field
    Set character limit of childName_field to 20
```

// For background music script

```
function Start()
    if instance is null
        Set instance as this script instance
        Make this game object persistent across scenes
    else
        Destroy this game object and return
    if device parameters are not available
        Scan and retrieve device parameters
    if music source of instance is null
        Instantiate the music prefab
        Retrieve the AudioSource component from the instantiated music prefab
and assign it to instance's music source
    Set the volume of the music source based on PlayerPrefs

function Update()
    if current scene is "EndStoryScene" or "ParentChooseEnvironmentScene"
        Pause the music source
    else if current scene is "AudioSettingScene" or "Story2_2ClassroomScene"
or "Story2_1ClassroomScene" or "Story3_1ClassroomScene" or
"Story3_2PlaygroundScene"
```

```
        Resume the music source
    if close button of the VR device is pressed
        Pause the music source
```

// For audio voiceovers script

```
function Awake()
    Call CheckLanguageSelected function

function CheckLanguageSelected()
    Retrieve the language saved in PlayerPrefs
    If languageSaved is "English"
        Set the audio source's clip to audioClipEng
    Else if languageSaved is "Chinese"
        Set the audio source's clip to audioClipChi
    Else if languageSaved is "Melayu"
        Set the audio source's clip to audioClipMal
    Else
        Set the audio source's clip to audioClipEng
    Play the audio source
    Log a message indicating that audio is being played
```

// For sound setting script

```
function Start()
    Find the AudioSource component named "Happy-and-fun-background-
music" in the scene and assign it to musicSource
    If PlayerPrefs does not have the key "musicVolume"
        Set the "musicVolume" PlayerPrefs float to 1
        Call the Load function
    Else
        Call the Load function

function ChangeVolume()
    Set the volume of the musicSource AudioSource to the value of the
volumeSlider
    Call the Save function

function Load()
    Set the value of the volumeSlider to the float value stored in the PlayerPrefs
with the key "musicVolume"

function Save()
    Store the value of the volumeSlider in the PlayerPrefs with the key
"musicVolume"
```

// For VR mode script

```
function Start()
    Save the main camera from the scene as _mainCamera
    Configure screen settings to prevent sleep and set brightness to maximum
    Check if device parameters are stored, if not, scan them
    Get the current active scene
```

```
        Define arrays of scene names for VR mode and non-VR mode
        If the current scene name exists in the VRModeScenes array
            Call EnterVR() function
        Else if the current scene name exists in the NonVRModeScenes array
            Call ExitVR() function

    function Update()
        If the close button on the device is pressed
            Call the appropriate function(s) to handle the action (currently
    commented out)

    function EnterVR()
        Start the XR plugin asynchronously
        If new device parameters are available, reload them

    function ExitVR()
        Stop the XR plugin

    function StartXR()
        Initialize XR loader asynchronously
        If initialization is successful
            Start XR subsystems

    function StopXR()
        Stop XR subsystems
        Deinitialize XR loader
        Reset main camera's aspect ratio and field of view to default values

}
```

4.5 GUI Design

The images below are the GUI design for the VR application:

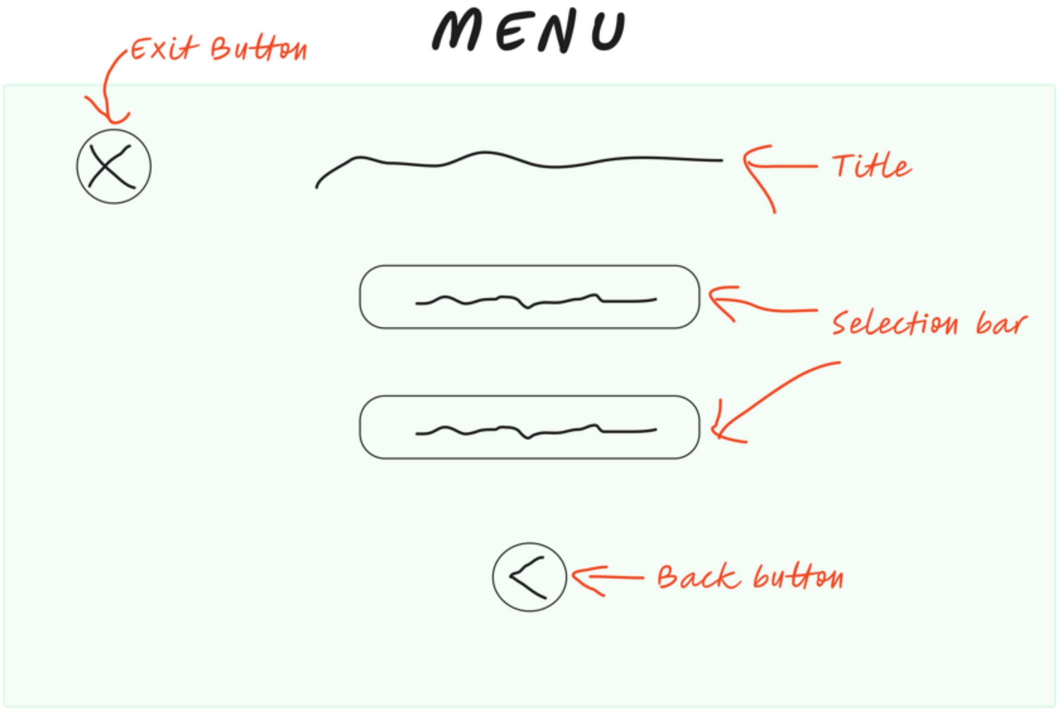


Figure 4.5.1 Wireframe for Menu

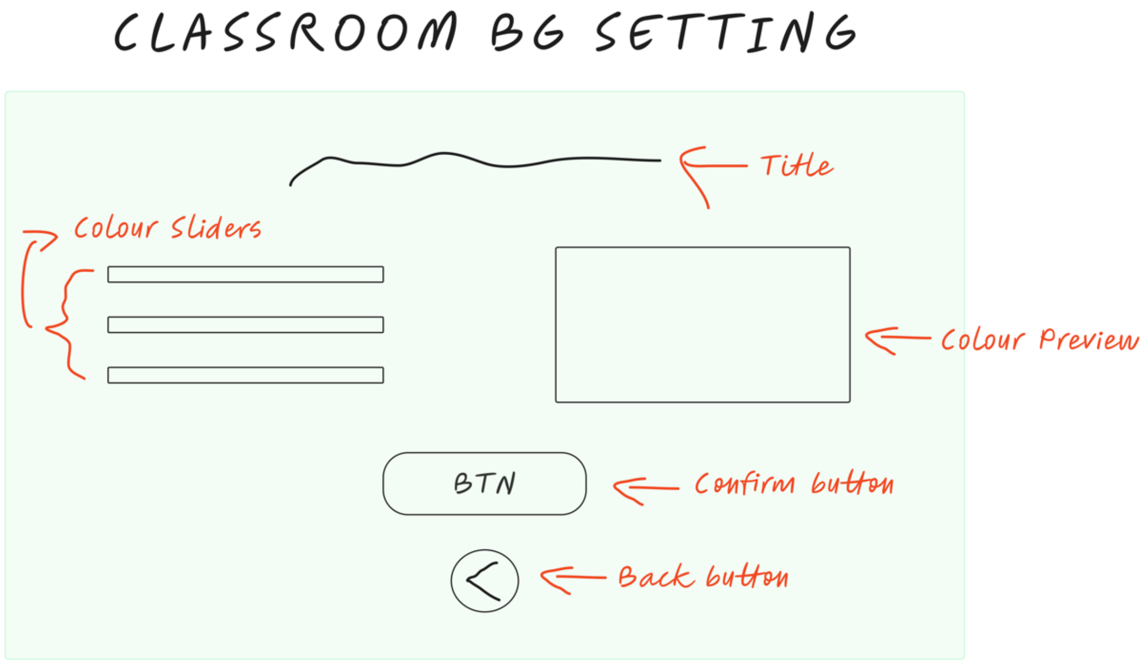


Figure 4.5.2 Wireframe for Classroom Background Setting

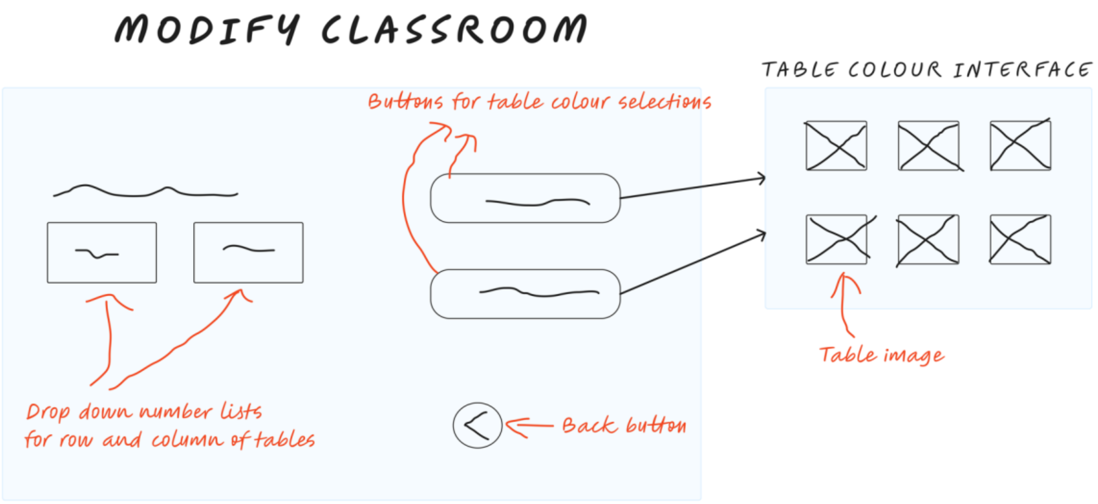


Figure 4.5.3 Wireframe for Modify Classroom Interface

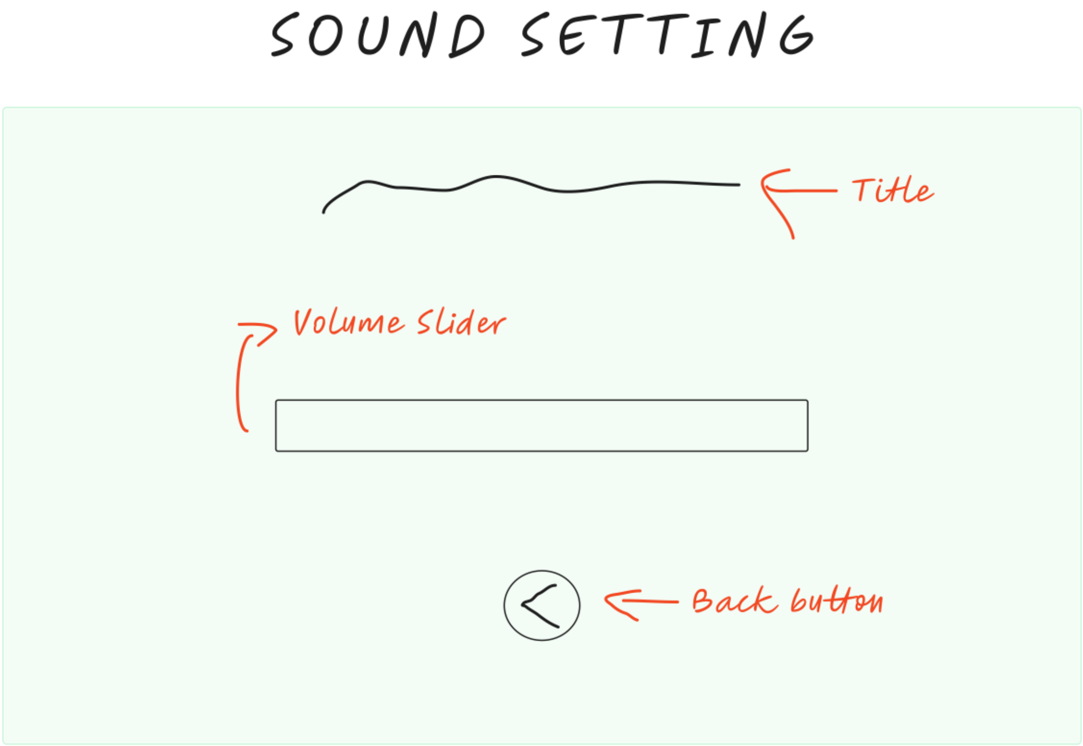


Figure 4.5.4 Wireframe for Sound Setting

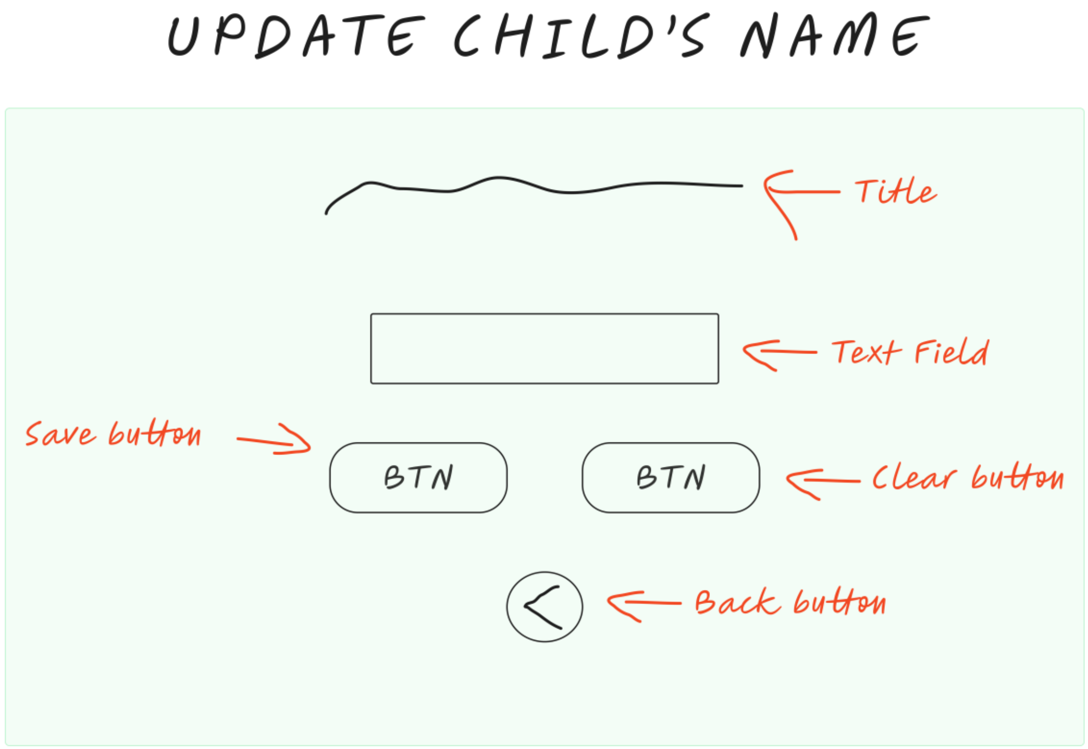


Figure 4.5.5 Wireframe for Update Child's Name

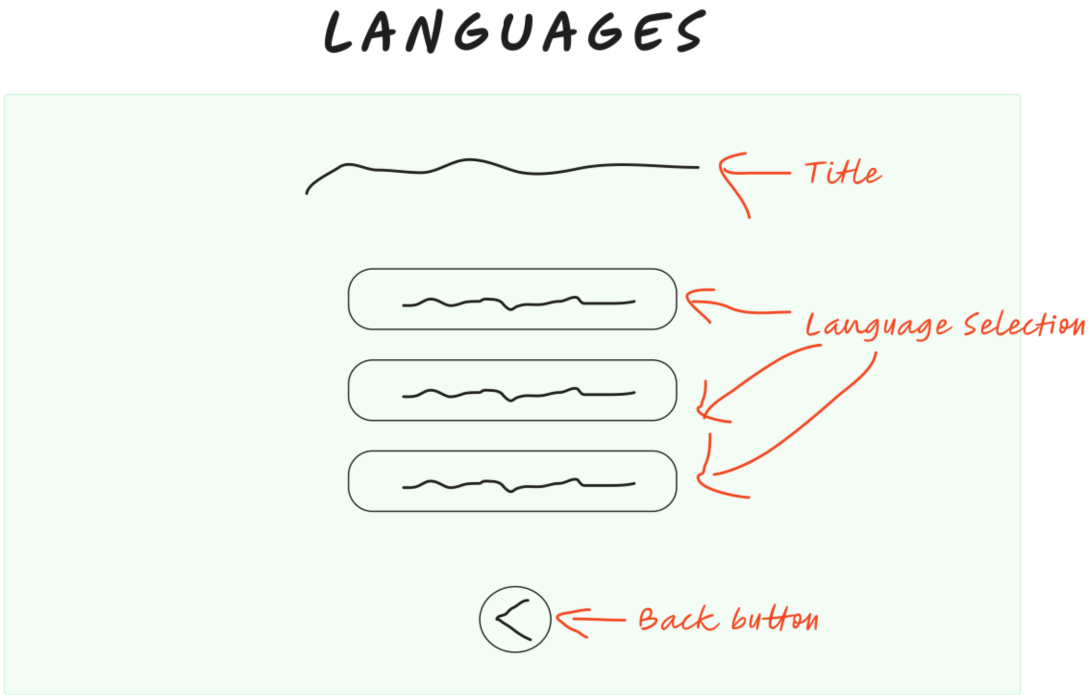


Figure 4.5.5 Wireframe for Language Selection

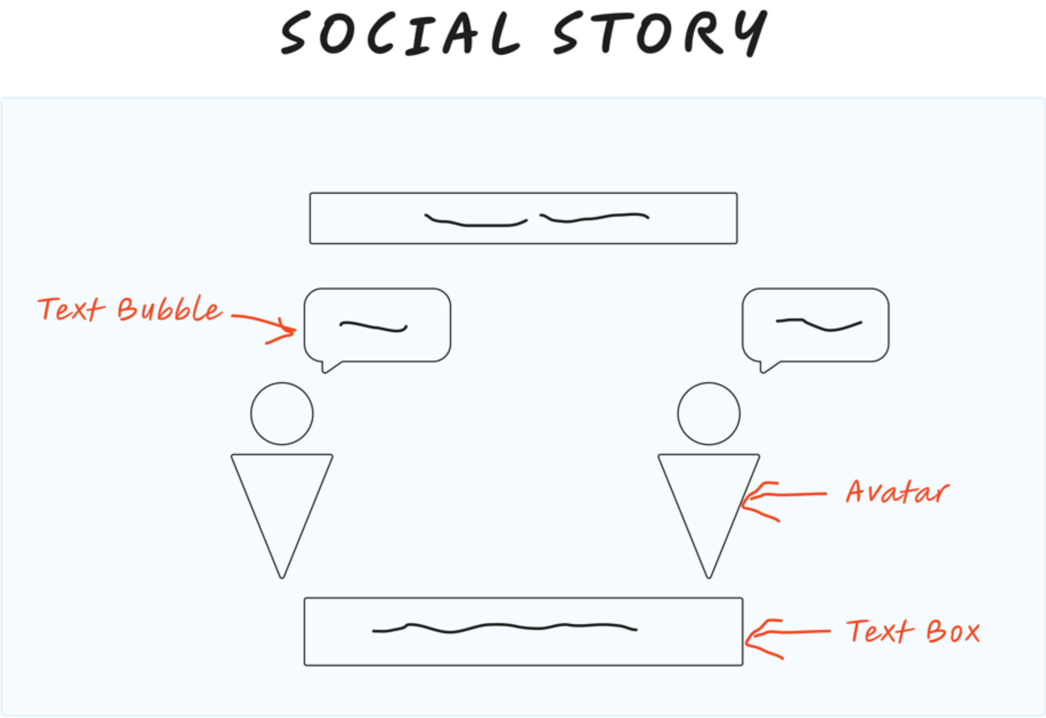


Figure 4.5.7 Wireframe for Social Story Scene

4.6 Avatar Design

The VR application requires careful consideration of both simplicity and complexity in the design of the avatars, both the teacher and the student. The following provides a thorough explanation of the design process and the complexity hidden behind the seemingly straightforward appearances of the avatars:

Student avatar:

The student avatar's design is intentionally simple, with a blue long-sleeved blouse and black pants that serve as a broadly familiar and simply customizable depiction of the student's real-world outfit. However, the complexity comes from its functional depth. The avatar is designed with appropriate body proportions and a neutral skin tone to prevent excessive characteristics that may overwhelm autistic students. The hands have been carefully developed to allow for natural and intuitive interactions within the VR environment, making it easier to practice social gestures and cues. Additionally, the absence of facial characteristics is a deliberate choice intended to avoid discomfort or sensory overload caused by interpreting facial expressions.

Figure 4.6.1 Student Avatar Design

Teacher avatar:

The teacher avatar is deliberately created to provide a familiar and relaxing presence to autistic students. The low-polygon 3D rendering approach attempts a careful balance between delivering a visually appealing aesthetic and avoiding the sensory overload that more complicated, high-polygon models may produce. The teacher avatar's casual outfit, which includes a yellow coat with red cuffs, blue pants, and brown shoes, id purposefully designed to distinguish the figure as a teacher while keeping a pleasant, non-threatening style. The gentle, harmonizing colour scheme reveals an in-depth understanding of autistic students' preferences, with the goal of creating a relaxing effect and a pleasant learning atmosphere.

Figure 4.6.2 Teacher Avatar Design

The complexity of both avatars comes from their flexibility and careful balance of simplicity and detail. Each component is carefully chosen to adapt to autistic students' sensitivities and demands, resulting in an interesting and engagement learning experience. The flexibility ensures that the virtual environment is closely aligned with the student's real-world classroom setting, increasing the sense of familiarity and comfort. Moreover, the avatars' look reveals the complexity of the underlying design process. Extensive study and thought were put into understanding autistic students' particular requirements and preferences, ensuring that each characteristic of the avatars contributes to establishing a pleasant and non-threatening virtual environment that promotes successful social skill acquisition.

4.7 Concluding Remark

This chapter will describe the system architecture of this project in the form of block diagram. The functional modules in the VR system are listed in the table so that it is more readable. This chapter also will show the system flow in the form of flowchart. In addition to that, the algorithm design is included in this chapter to show how the C## scripts interact with the elements in Unity. The chapter also will reveal the GUI design of this project in the form of wireframe. This chapter will highlight the complexity of avatar design.

CHAPTER 5

System Implementation

5.1 Software Setup

5.1.1 Unity Setup

1. Go to website https://unity.com and click on “PLANS AND PRICING” button.

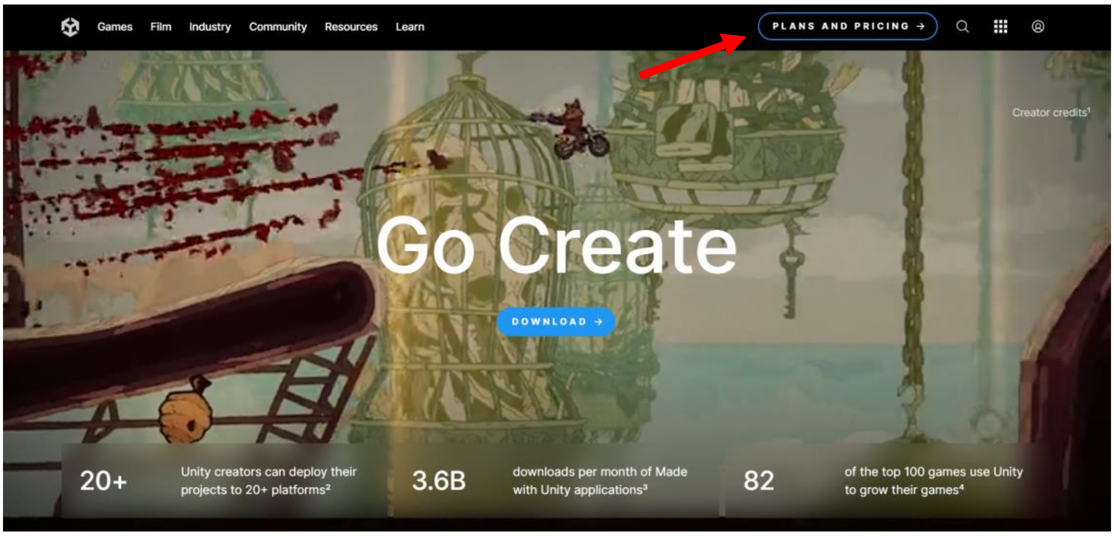


Figure 5.1.1.1 Unity Official Website

2. Select the “Student and hobbyist” bar.

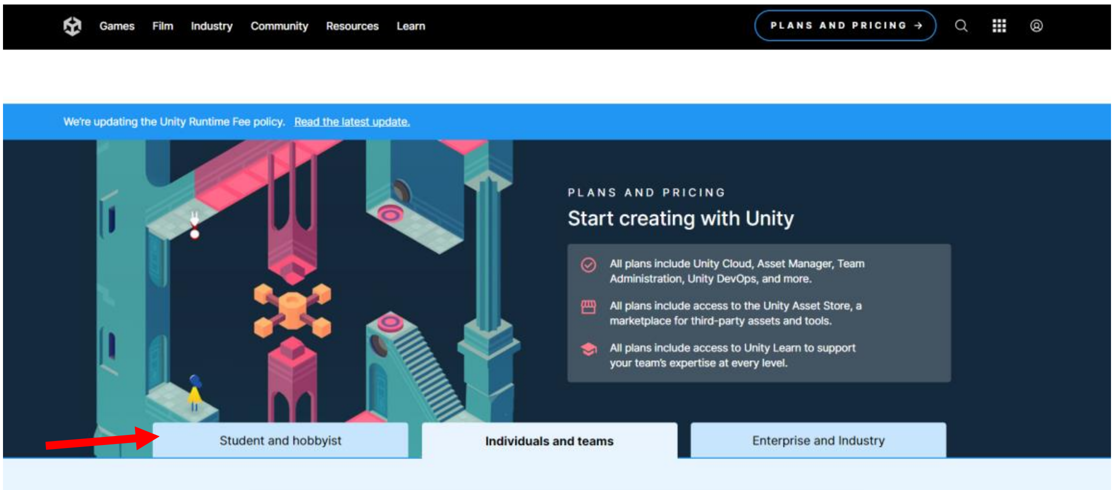


Figure 5.1.1.2 Unity Plan and Pricing

3. Click on “Sign up” button.

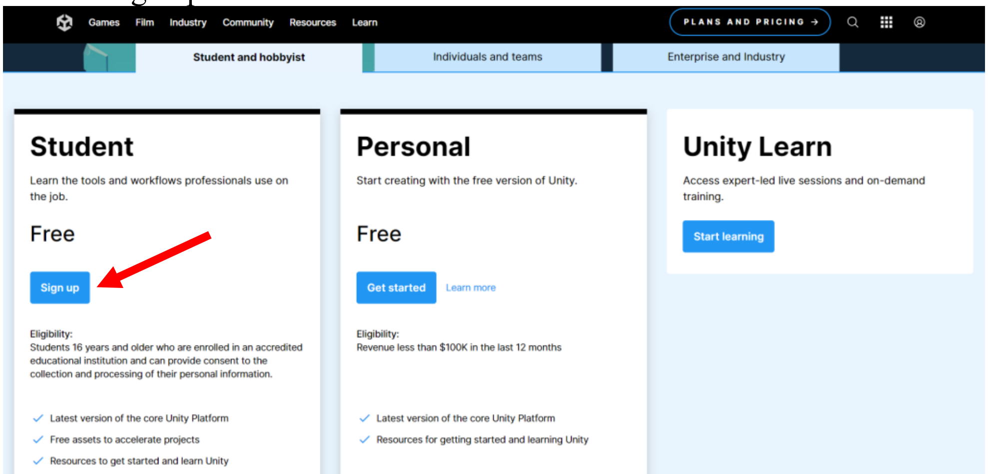


Figure 5.1.1.3 Student and Hobbyist Plan

4. Click on “Free access: Post-secondary” button.

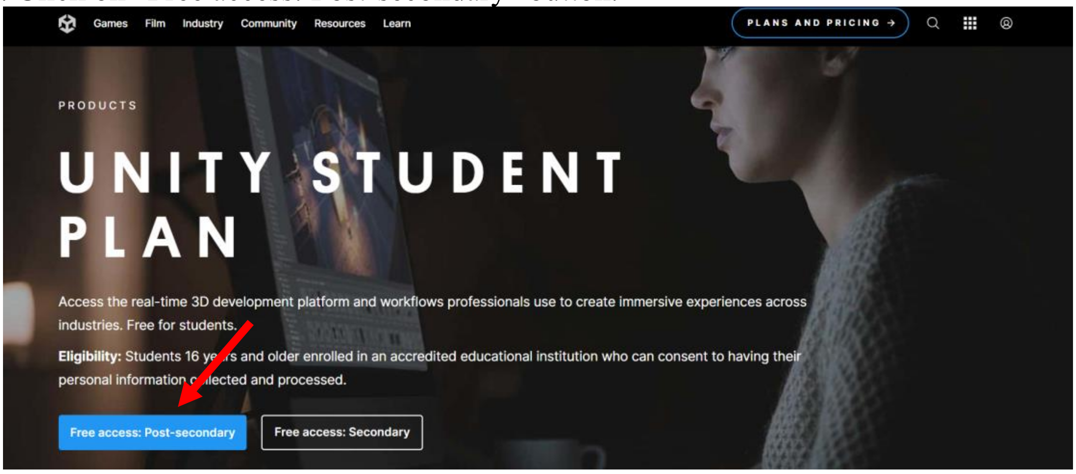


Figure 5.1.1.4 Student Plan

5. Sign in unity account or create a new account if does not have one.

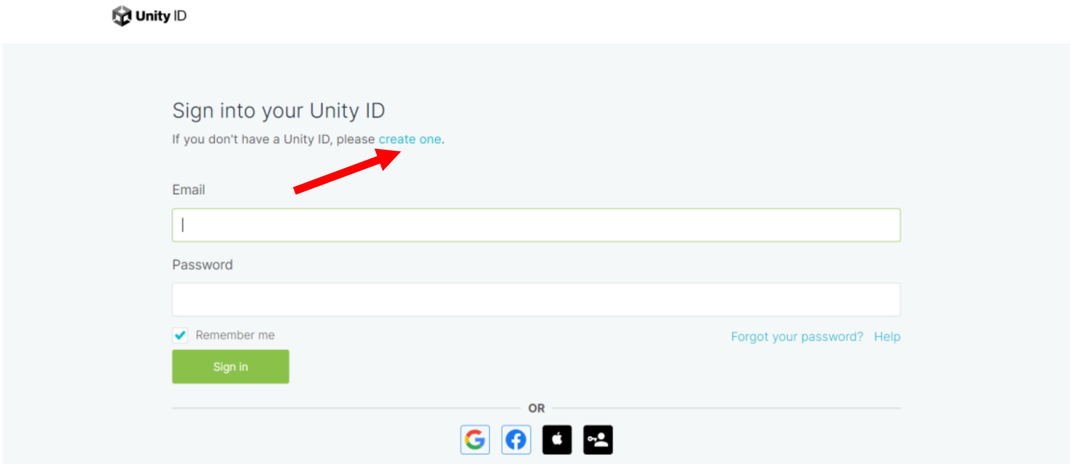


Figure 5.1.1.5 Sign in Page

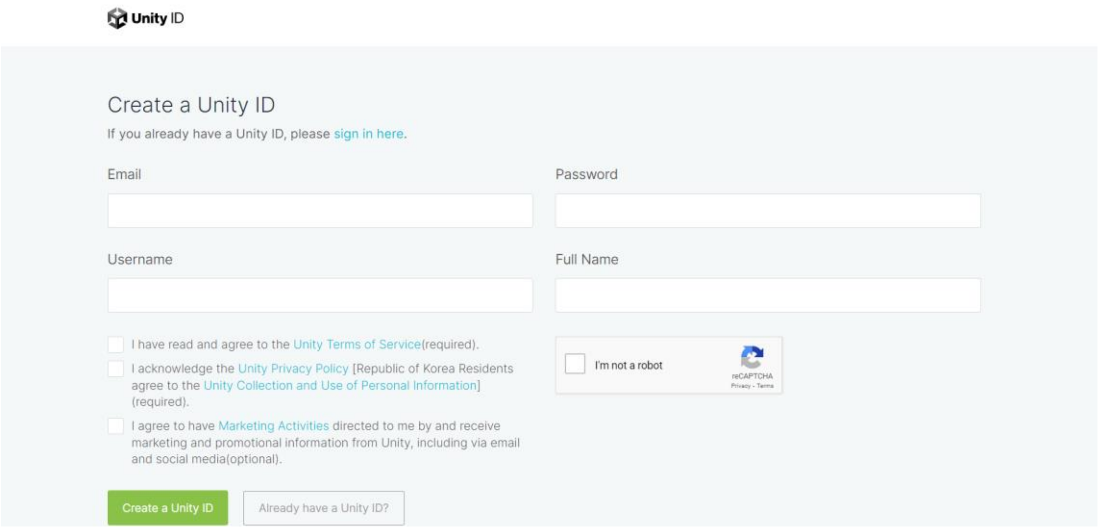


Figure 5.1.1.6 Create Account Page

6. Enter the required information and then click on “Verify My Student Status” button.

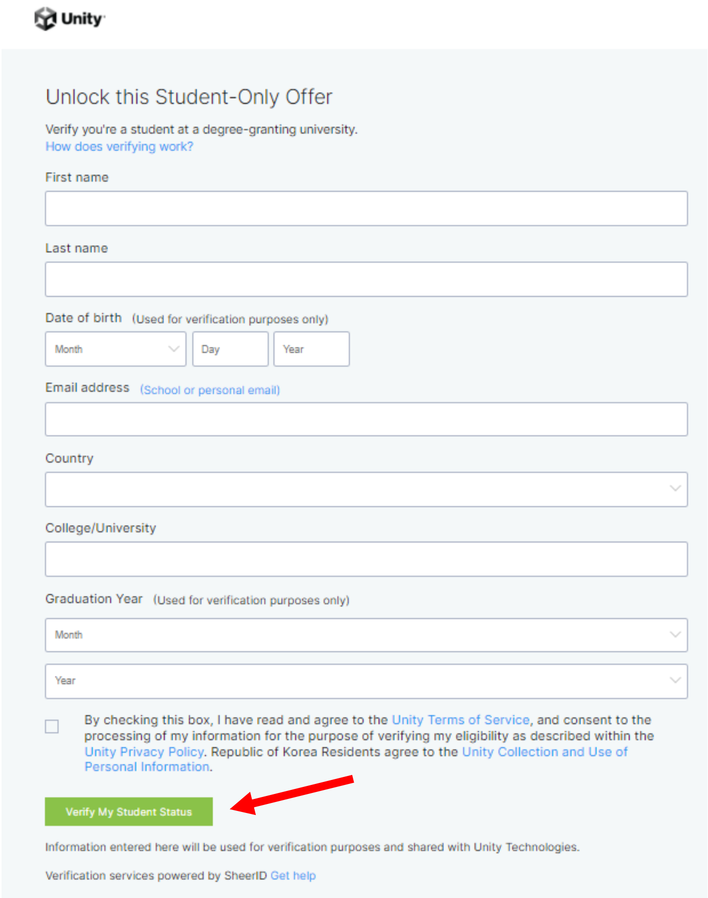


Figure 5.1.1.7 Verify Student Status Page

7. Click on “Sign in to my school” button to verify the school credentials.

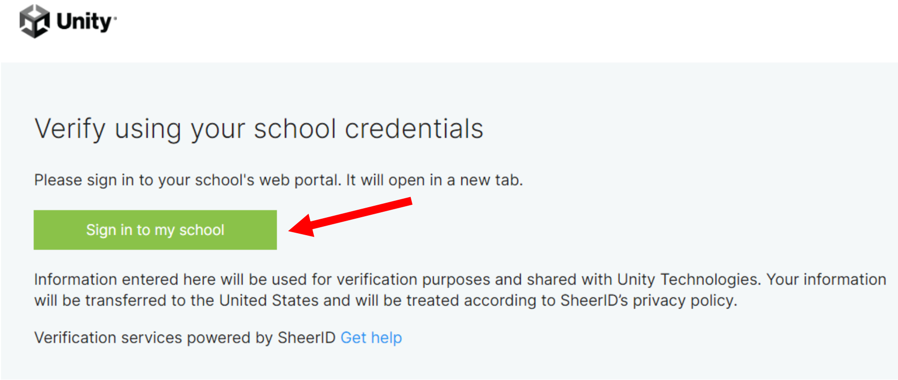


Figure 5.1.1.8 Verify School Credentials Page

8. After the verification, click on “Continue” button.

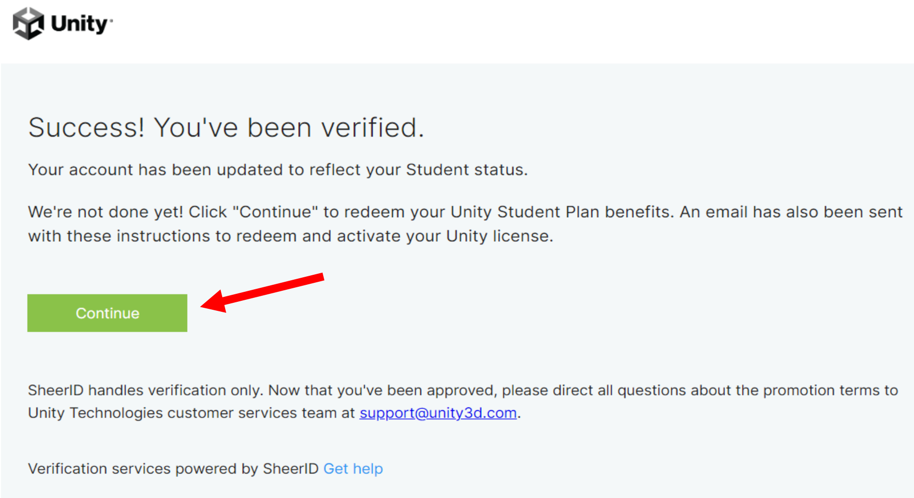


Figure 5.1.1.9 Verified Student Status Page

9. Scroll down the page and click the “Download Unity” link.

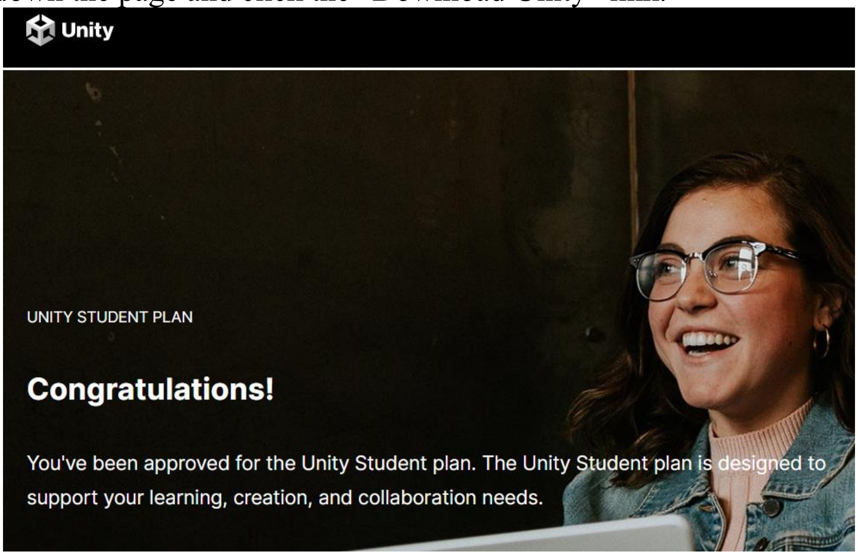


Figure 5.1.1.10 Verified Student Plan Page

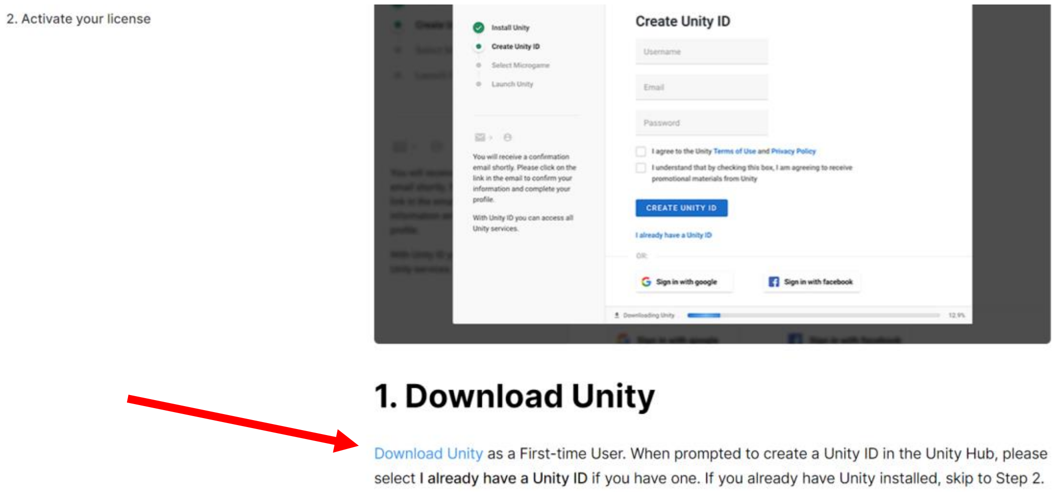


Figure 5.1.1.11 Download Unity Link

10. Click on “Download for Windows” then the installer file will be downloaded.

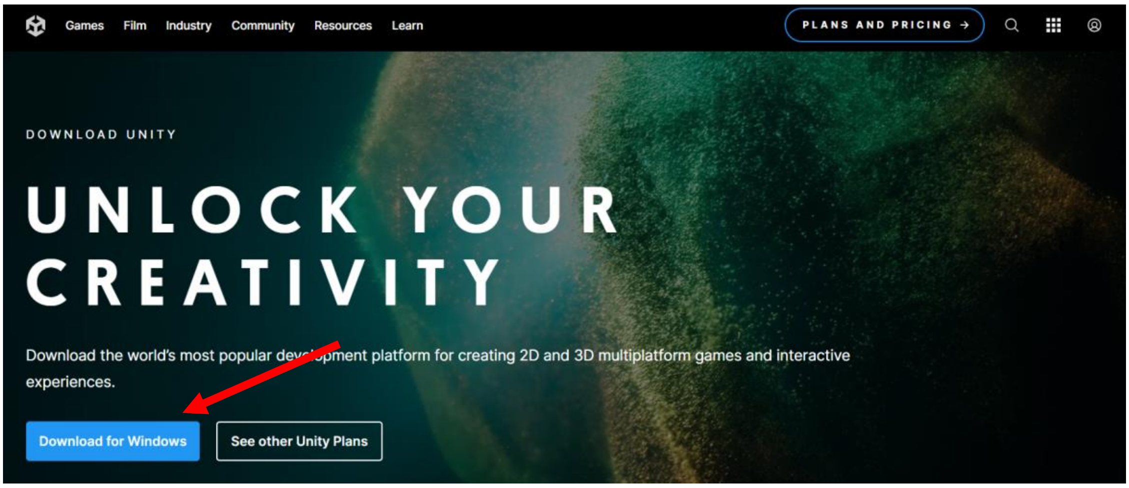


Figure 5.1.1.12 Download Unity Page

11. Execute the installer file.

12. Click on “I Agree” button.

Figure 5.1.1.13 License Agreement

13. Keep the default installation path and click on “Install” button.

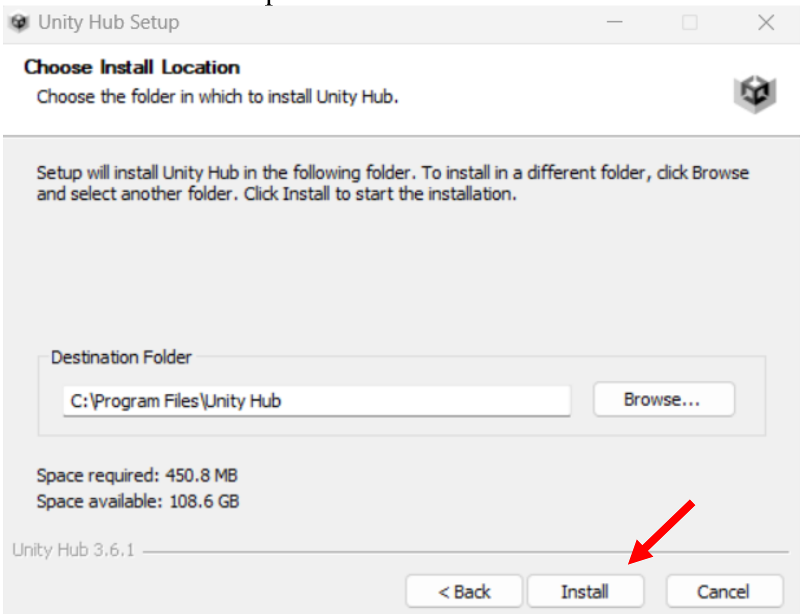


Figure 5.1.1.14 Choose Installation Path

14. After installation is completed, click on “Finish” button.

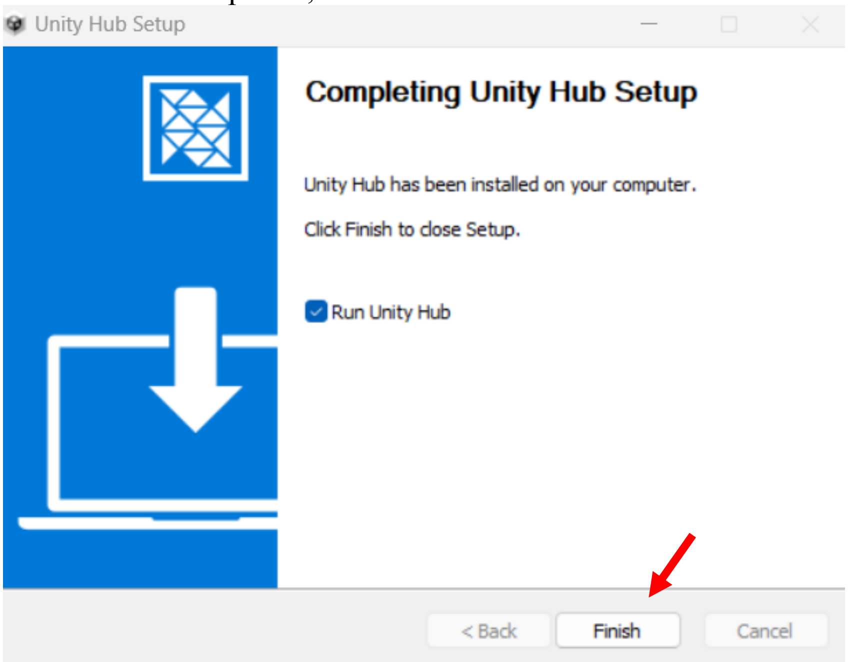


Figure 5.1.1.15 Installation Finished

15. Click on “Sign in” button. Then, sign in your Unity account.

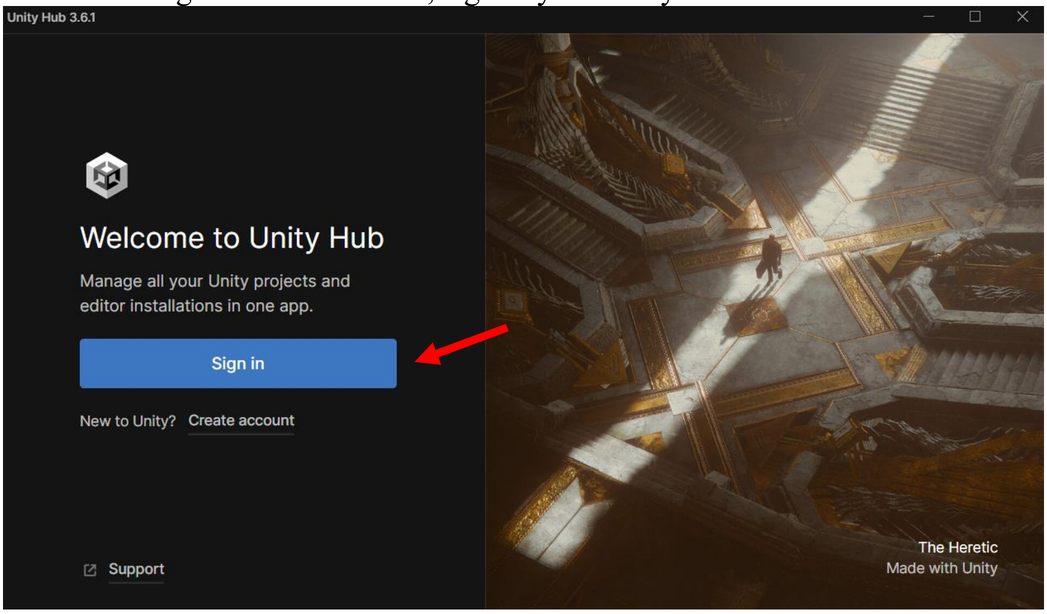


Figure 5.1.1.16 Sign in Unity Hub

16. Click on "Installs" section at the left side of the Unity Hub.

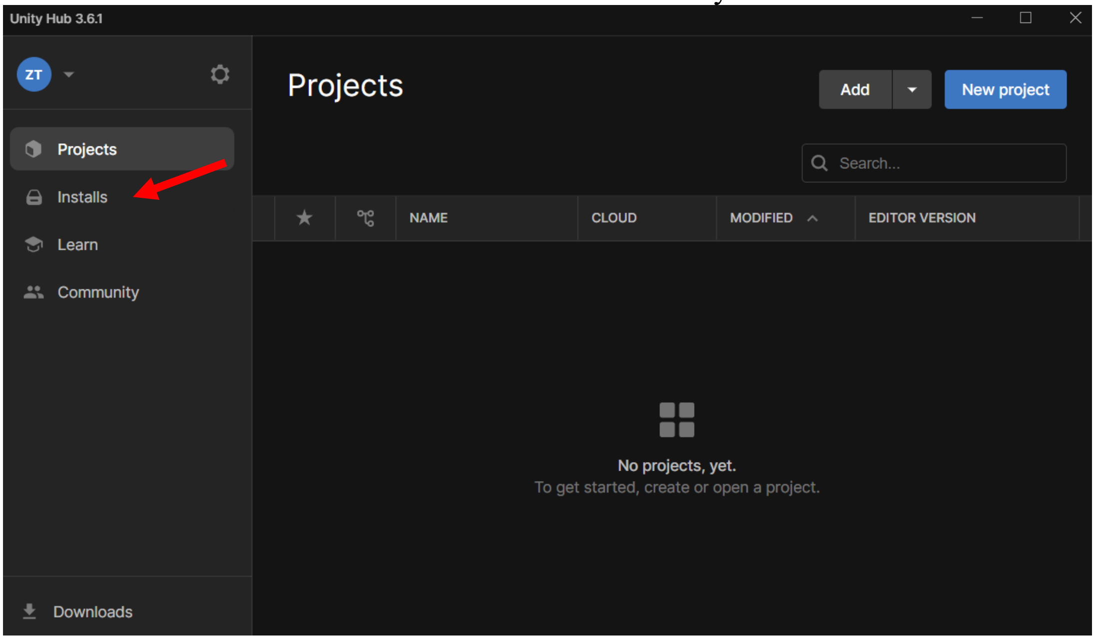


Figure 5.1.1.17 Unity Hub

17. Click on "Install Editor" button.

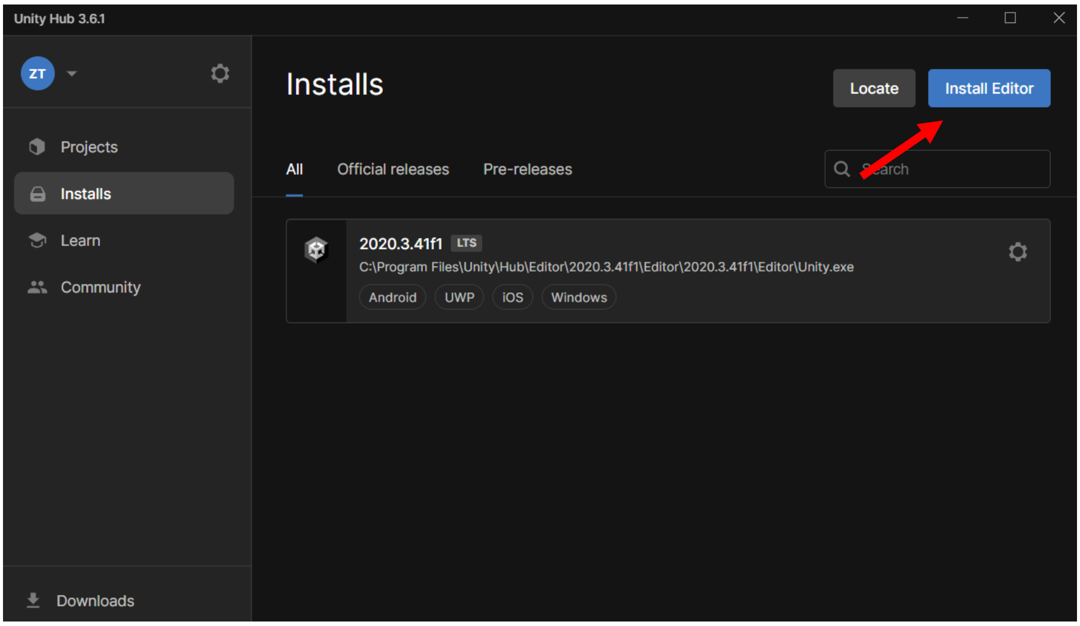


Figure 5.1.1.18 Installs Section

18. Select the Unity Editor version that desired and click "Install" button.

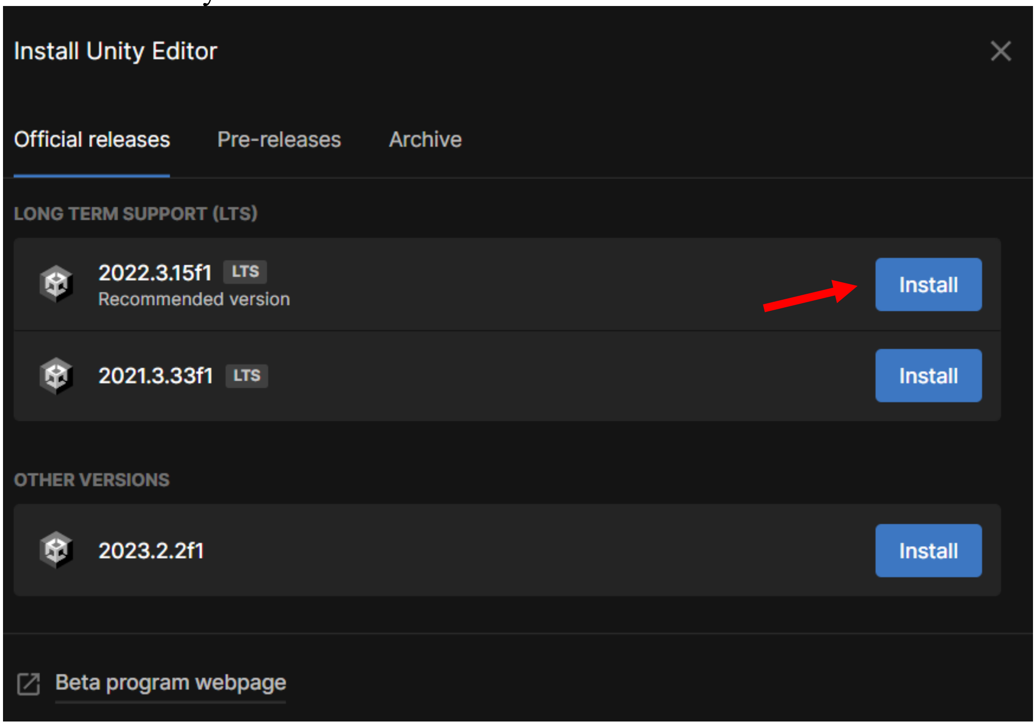


Figure 5.1.1.19 Install Unity Editor

19. Add the modules that required, then click "Continue" button.

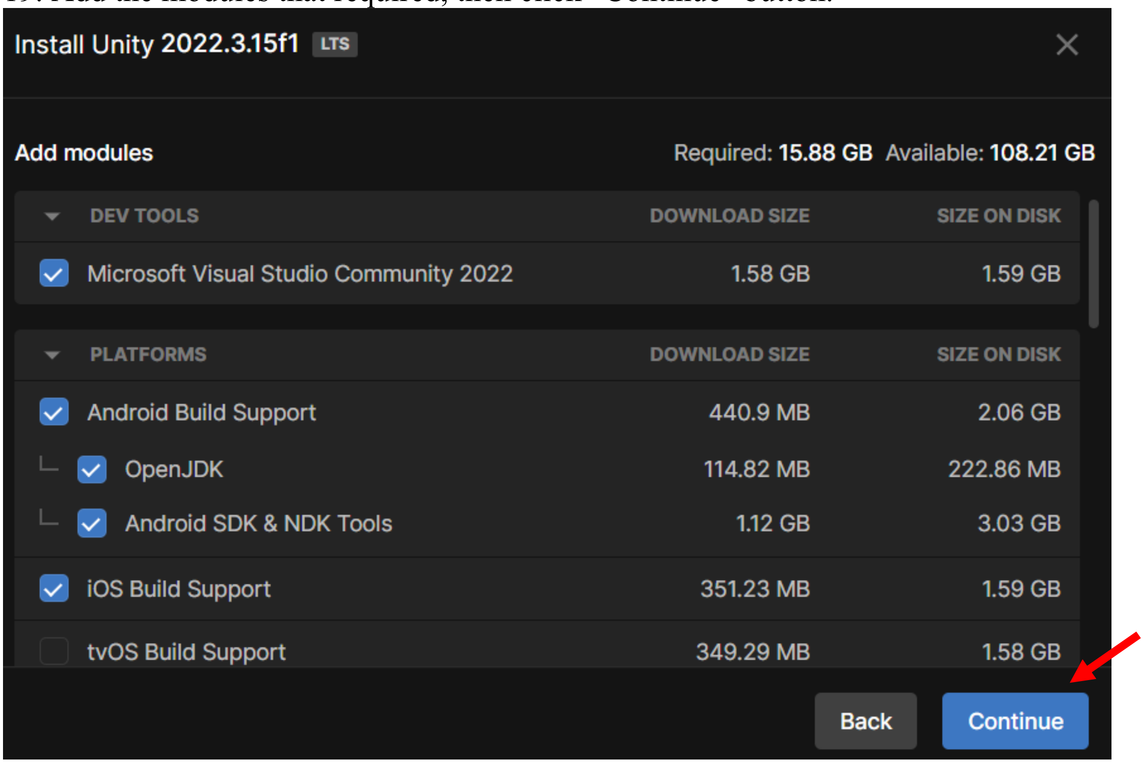


Figure 5.1.1.20 Add Modules

20. After Unity Editor is installed, go back to the Projects section, then click on "New project" button to create new project.

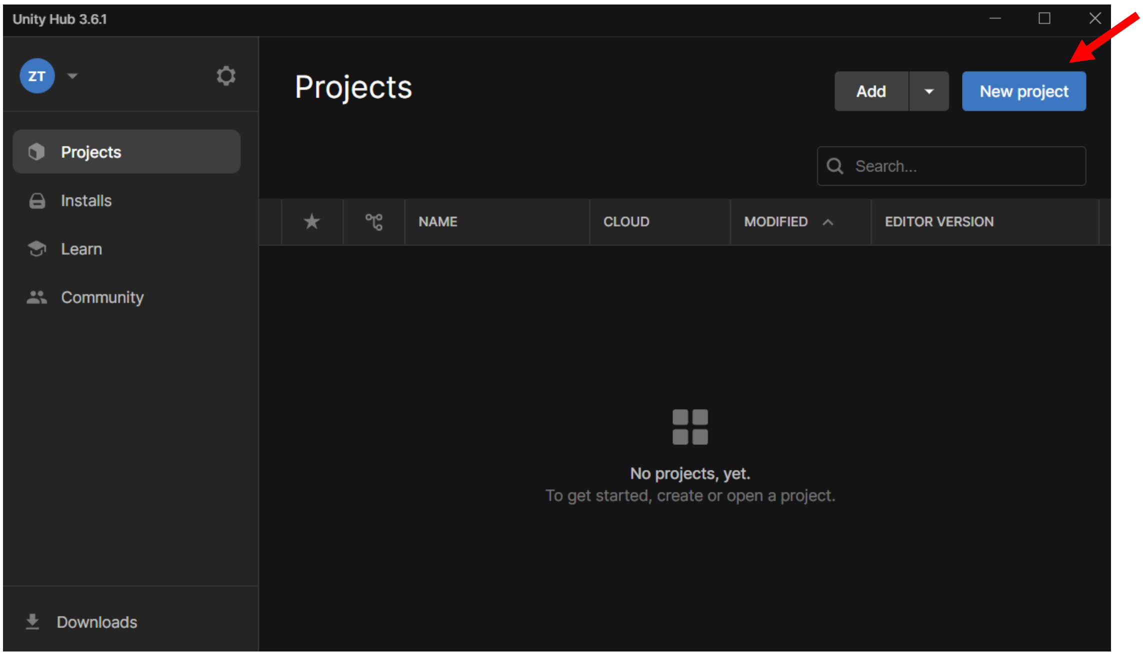


Figure 5.1.1.21 Projects Section

21. Select 3D template, then you may rename the project name. After that, click on “Create project” button.

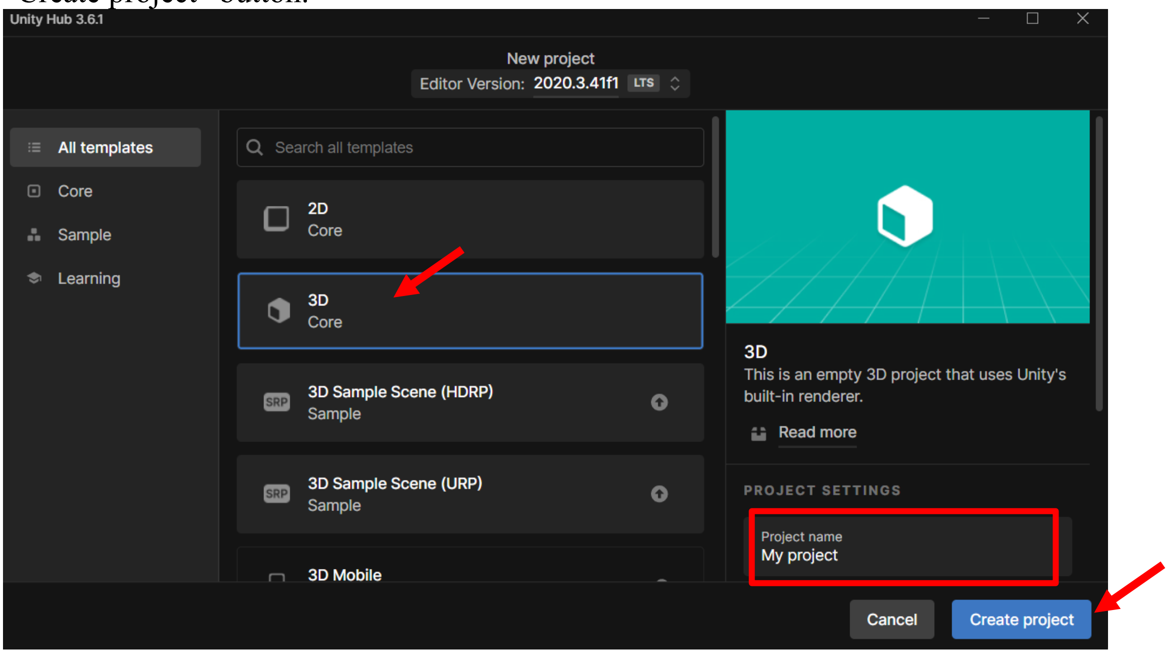


Figure 5.1.1.22 New Project

22. The new project will launch in Unity Editor.

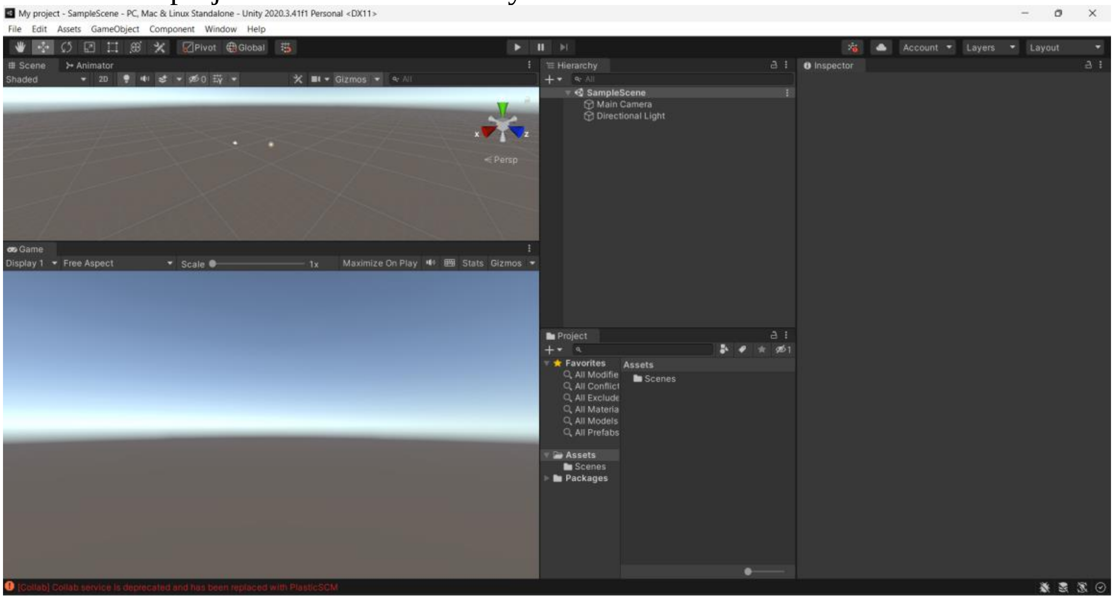


Figure 5.1.1.23 Unity Editor

5.1.2 Microsoft Visual Studio Setup

1. Go to website https://visualstudio.microsoft.com and click on "Downloads" button.

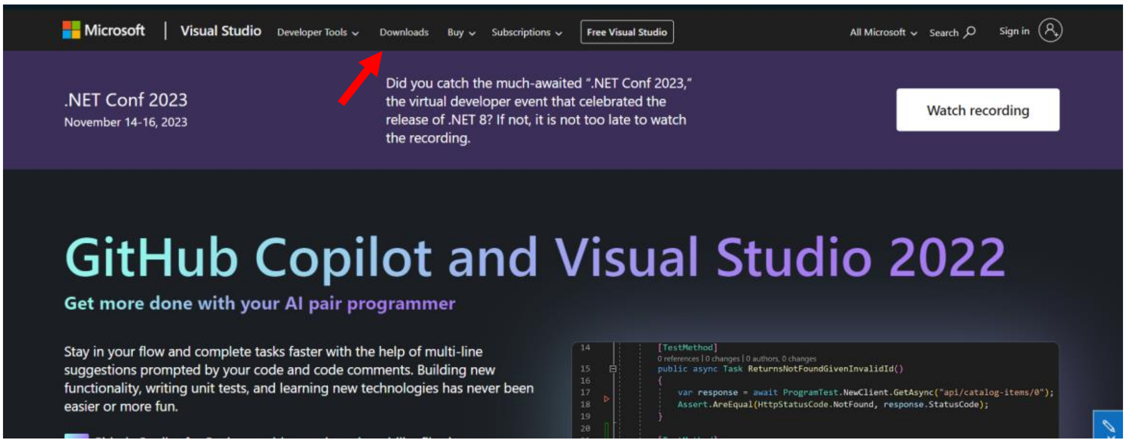


Figure 5.1.2.1 Microsoft Visual Studio Official Website

2. Click on "Free download" button at the Community section, then wait for the installer file to be downloaded.

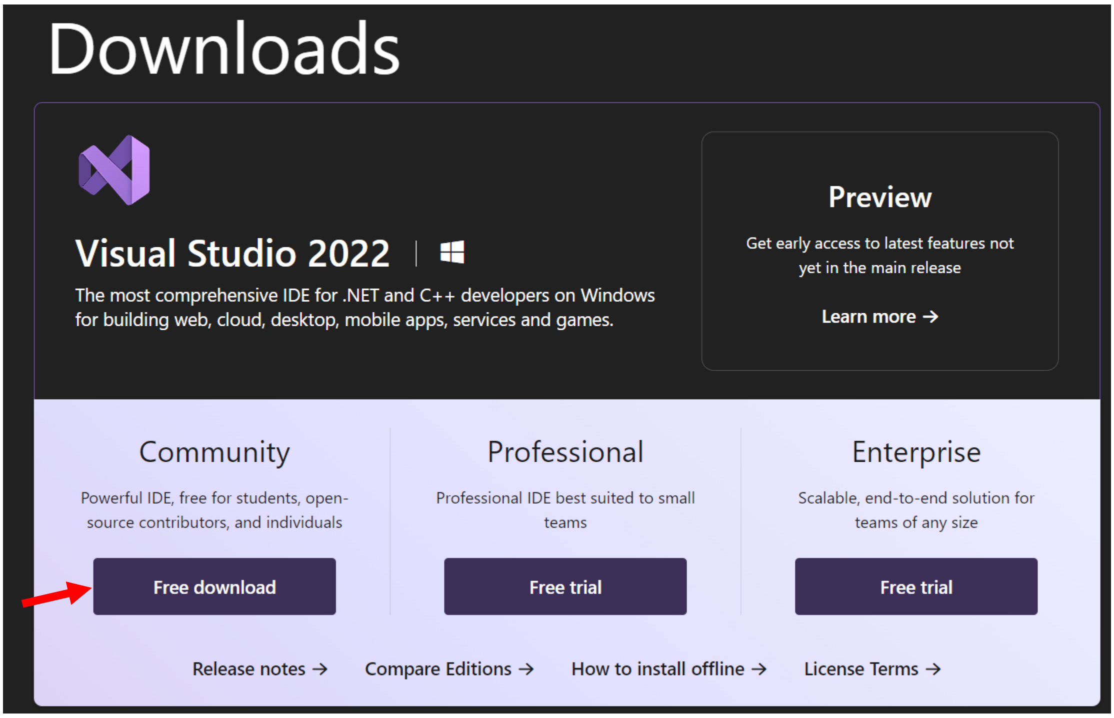


Figure 5.1.2.2 Microsoft Visual Studio Download Page

3. Execute the installer file.

4. Click on "Continue" button to setup the Visual Studio installer.

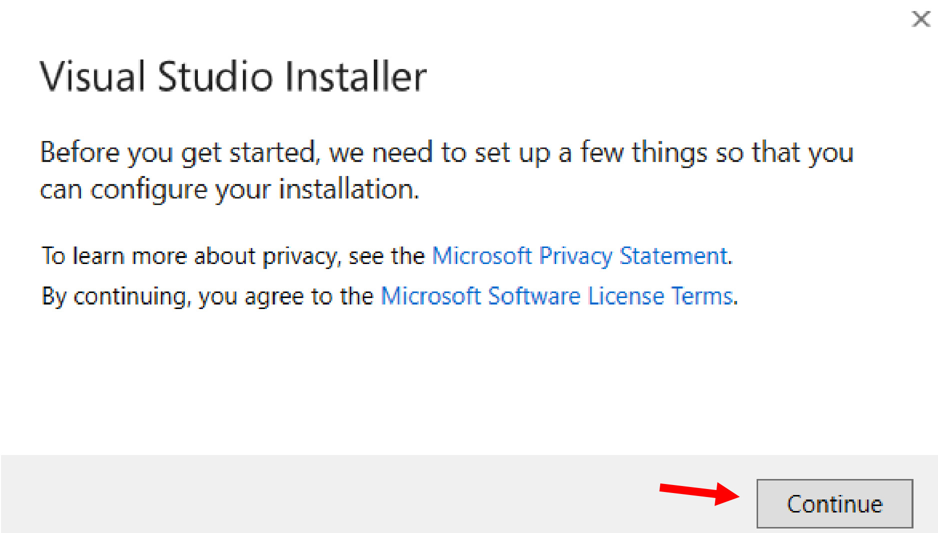


Figure 5.1.2.3 Setup Visual Studio Installer

5. Click on “Available” section at the top left of Visual Studio installer.

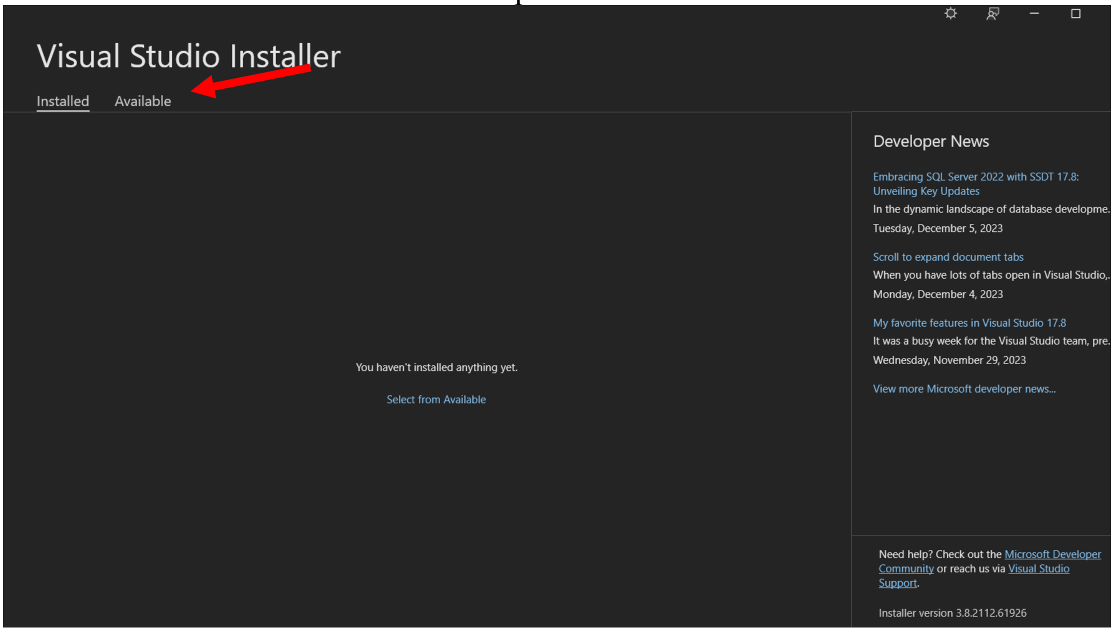


Figure 5.1.2.4 Visual Studio Installer

6. Click on “Install” button at the Visual Studio Community 2022.

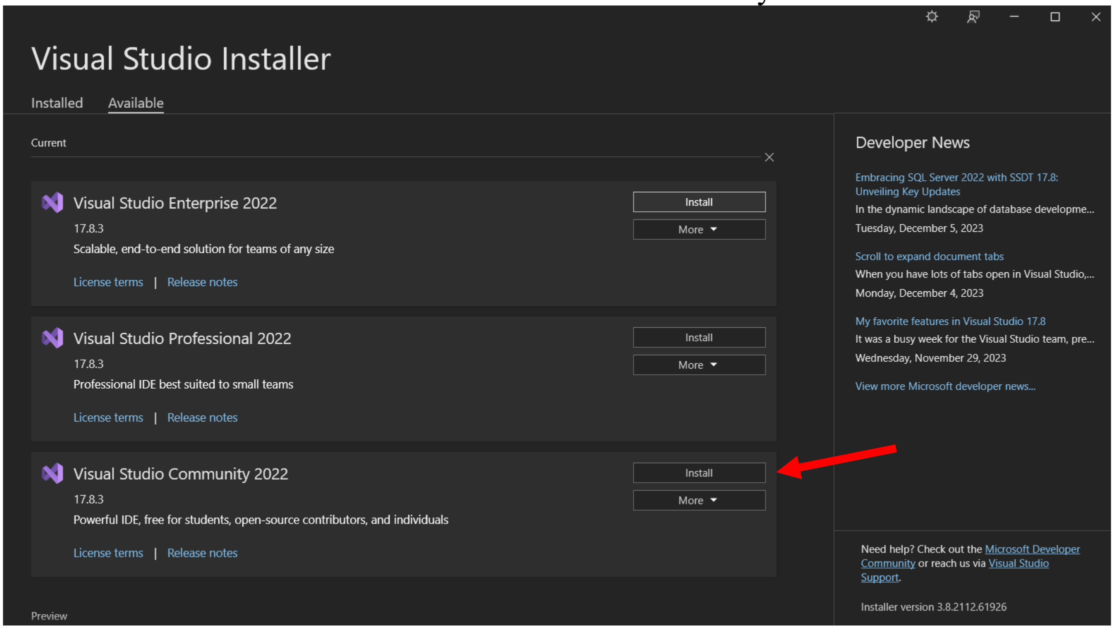


Figure 5.1.2.5 Install Visual Studio Community 2022

7. Add “Game development with Unity” at the Workloads tab, then click “Install” button.

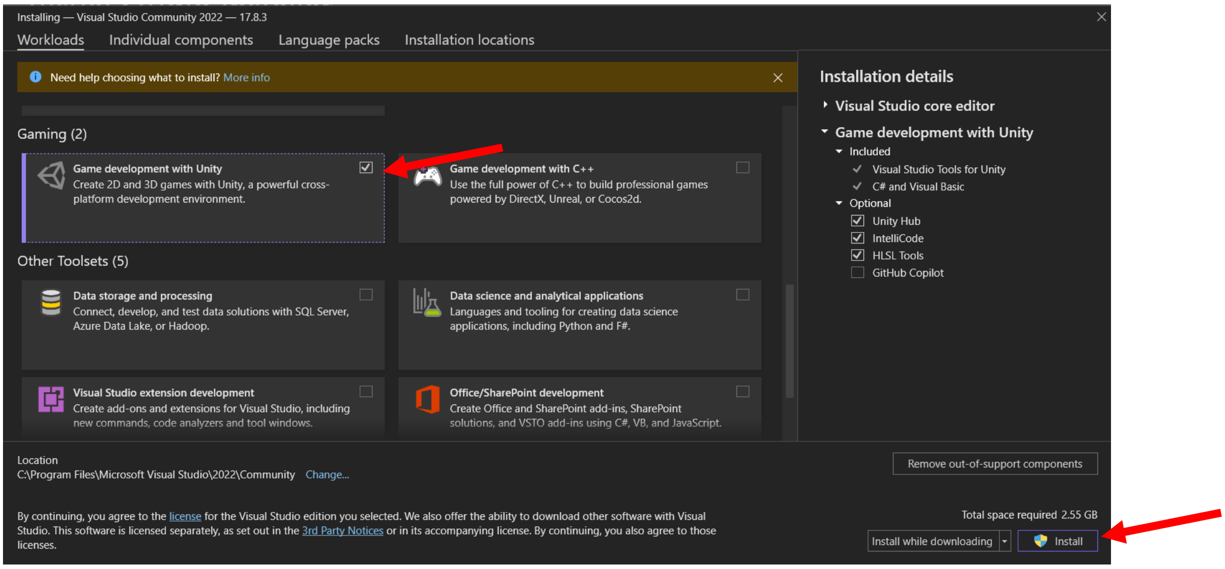


Figure 5.1.2.6 Workloads Tab

8. Click on “Sign in” button to sign in the Microsoft account.

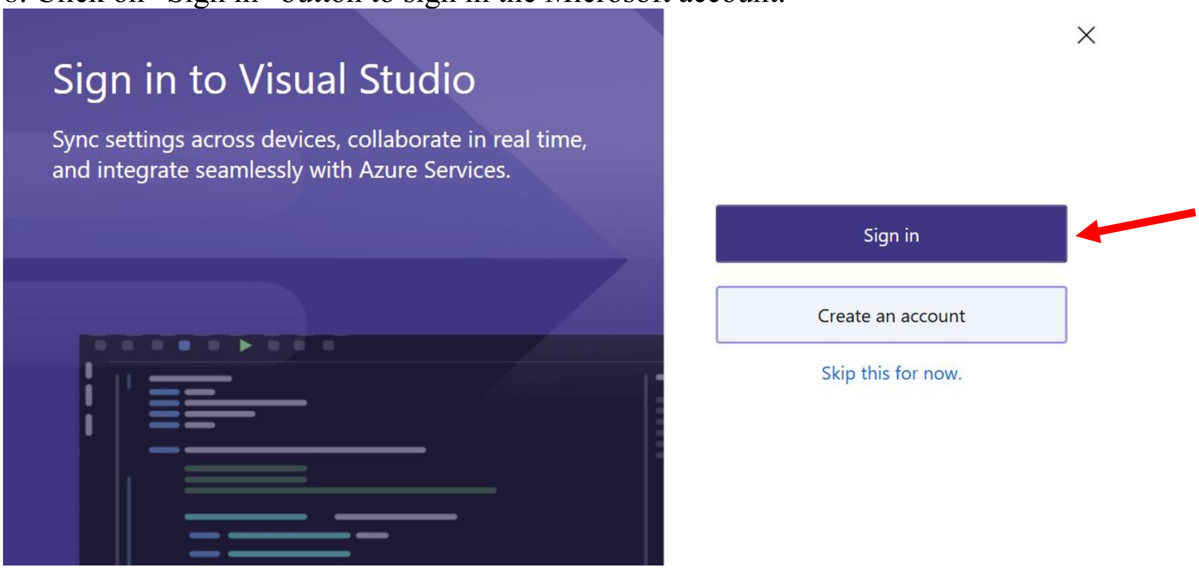


Figure 5.1.2.7 Sign in Visual Studio

9. Click on "Start Visual Studio" button.

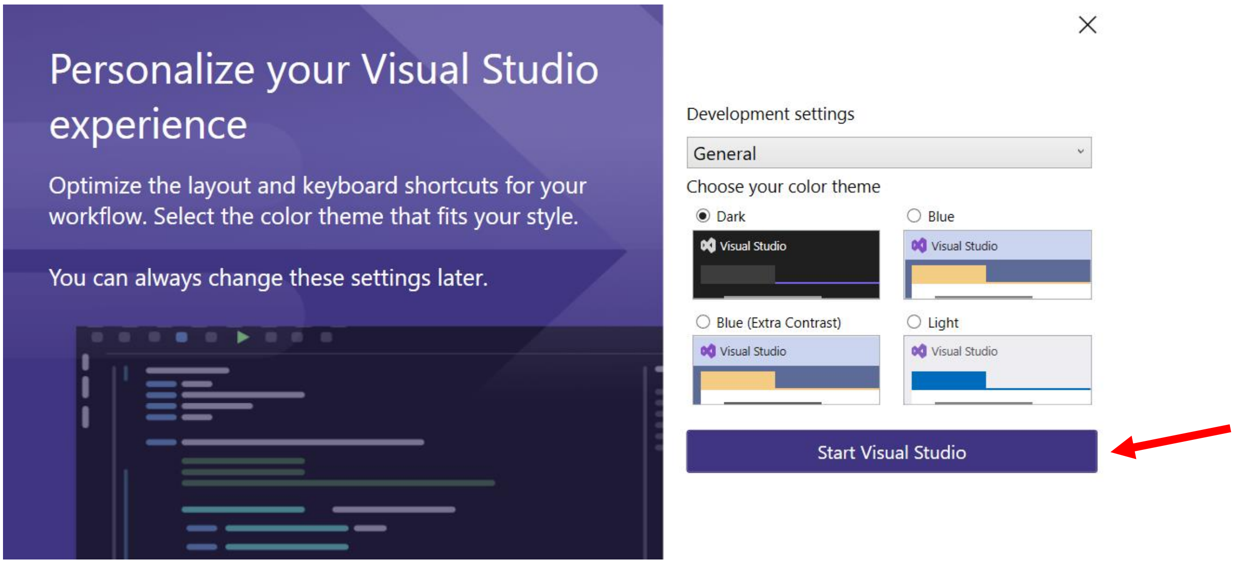


Figure 5.1.2.8 Start Visual Studio

10. Click on "Create a new project" button.

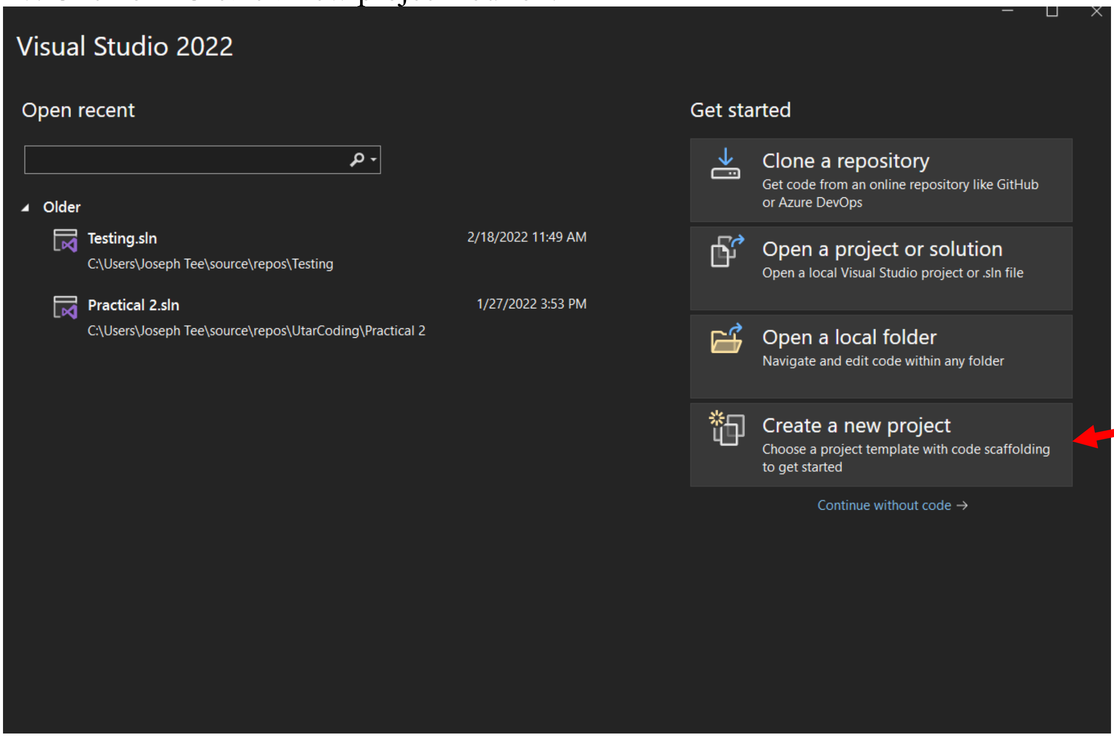


Figure 5.1.2.9 Create New Project

11. New Project will launch in Visual Studio Code.

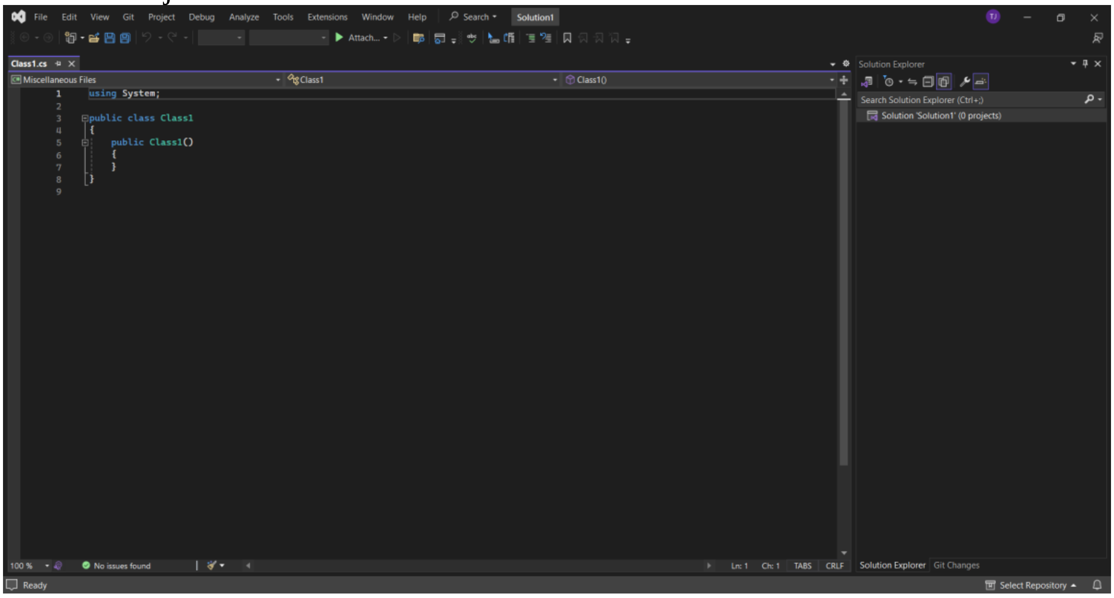


Figure 5.1.2.10 Visual Studio

5.1.3 Blender Setup

1. Go to Blender website: https://www.blender.org and click on “Download” button.

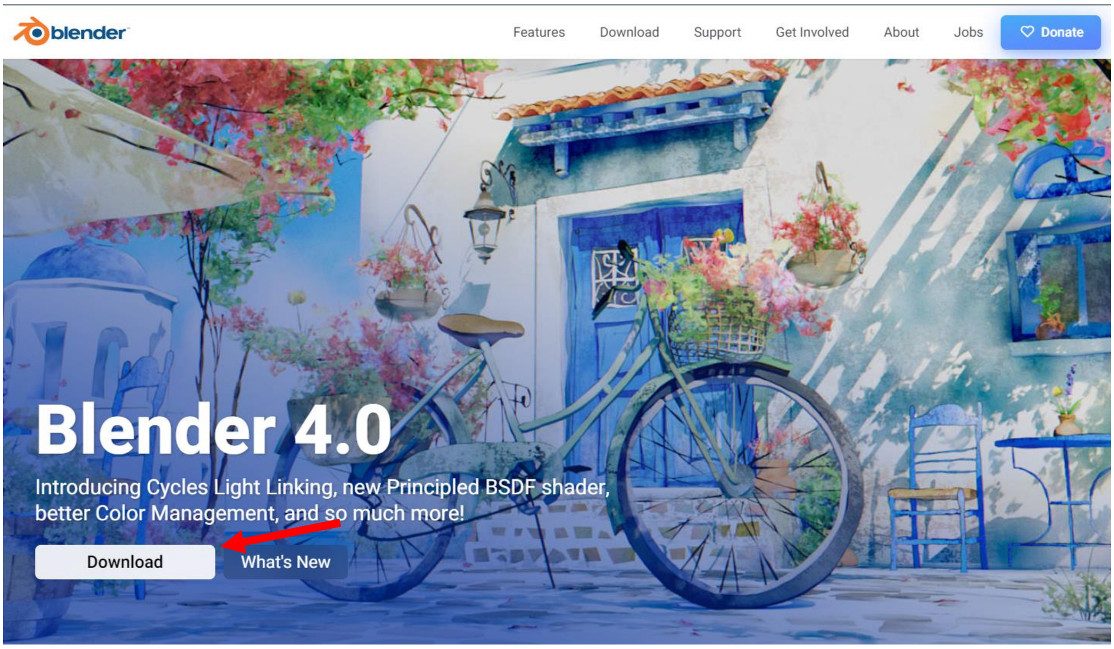


Figure 5.1.3.1 Blender Home Page

2. Click on “Download Blender 4.0.2” button.

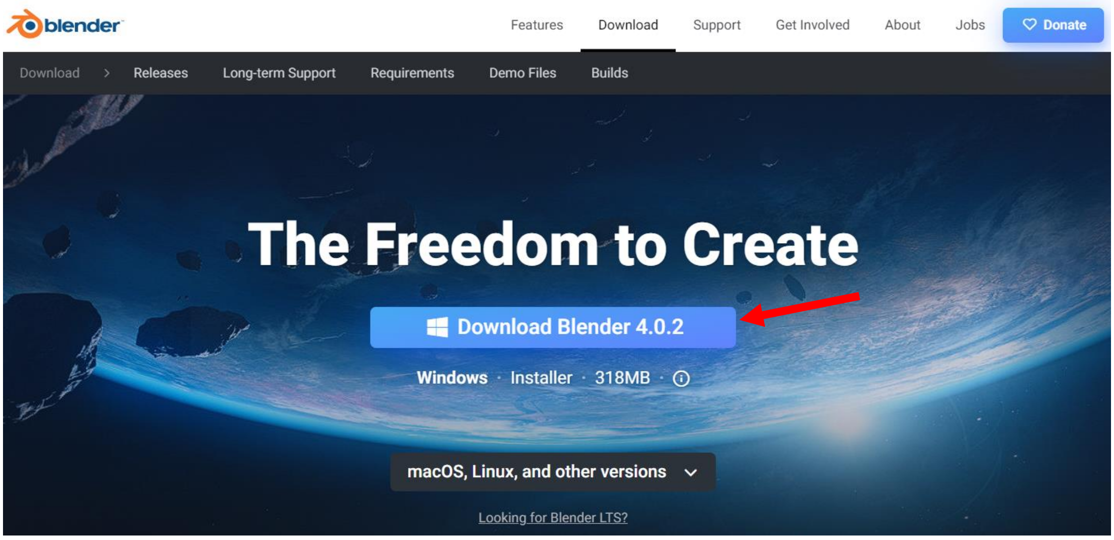


Figure 5.1.3.2 Blender Download Page

3. Wait for Blender installer file to be downloaded then execute the installer file.

4. Click on “Next” button.

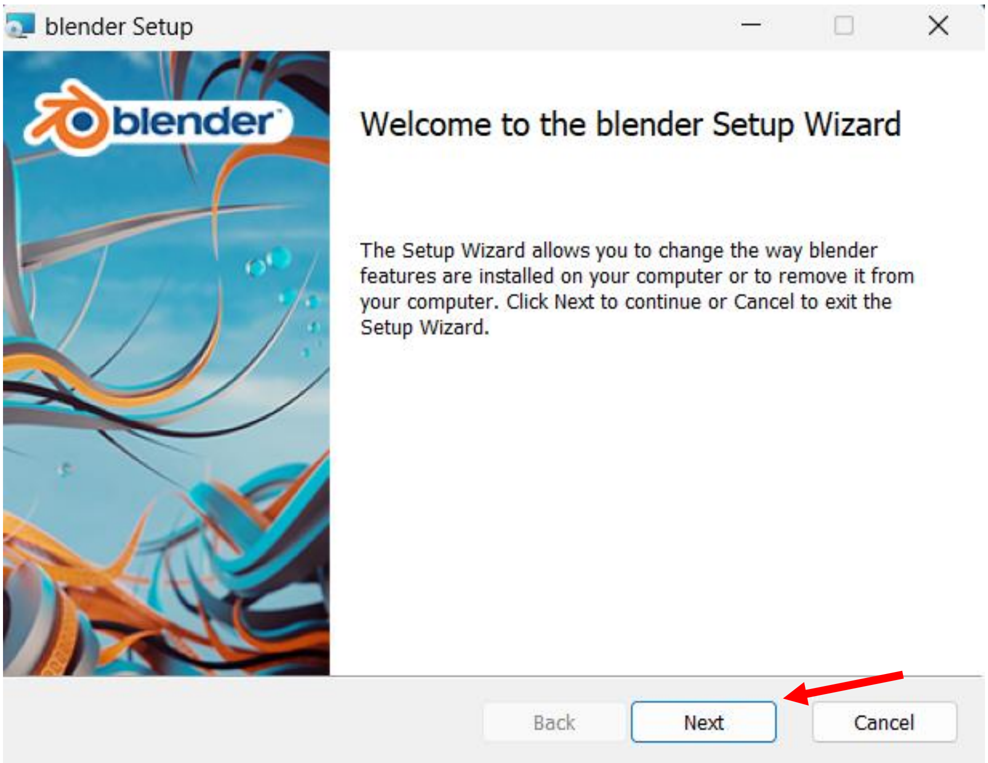


Figure 5.1.3.3 Blender Installer Setup

5. Click on checkbox to accept the agreement then click on “Next” button.

Figure 5.1.3.4 Blender License Agreement

6.Click on “Next” button.

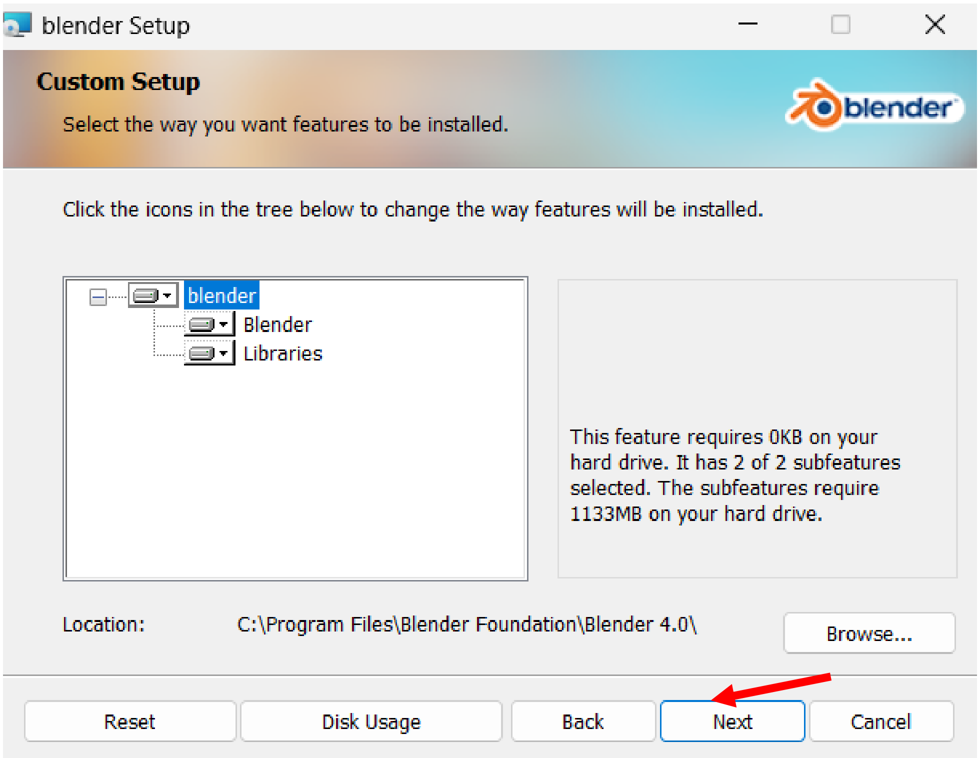


Figure 5.1.3.5 Installer Custom Setup

7. Click on "Install" button and wait the Blender to be installed.

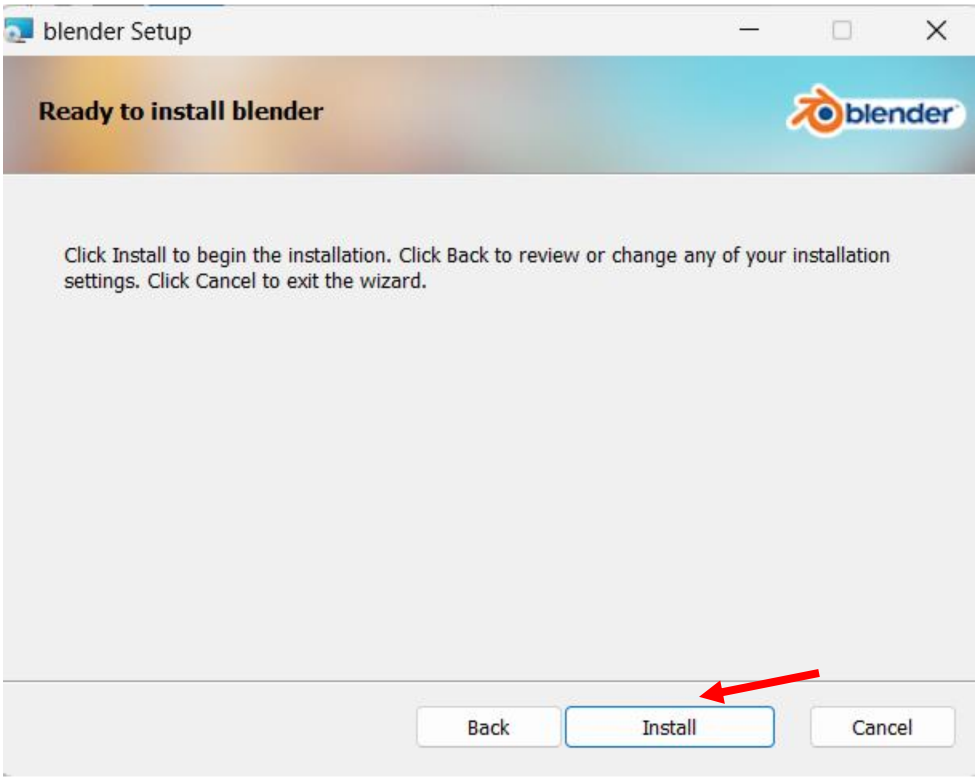


Figure 5.1.3.6 Install Blender

8. Click on "Finish" button.

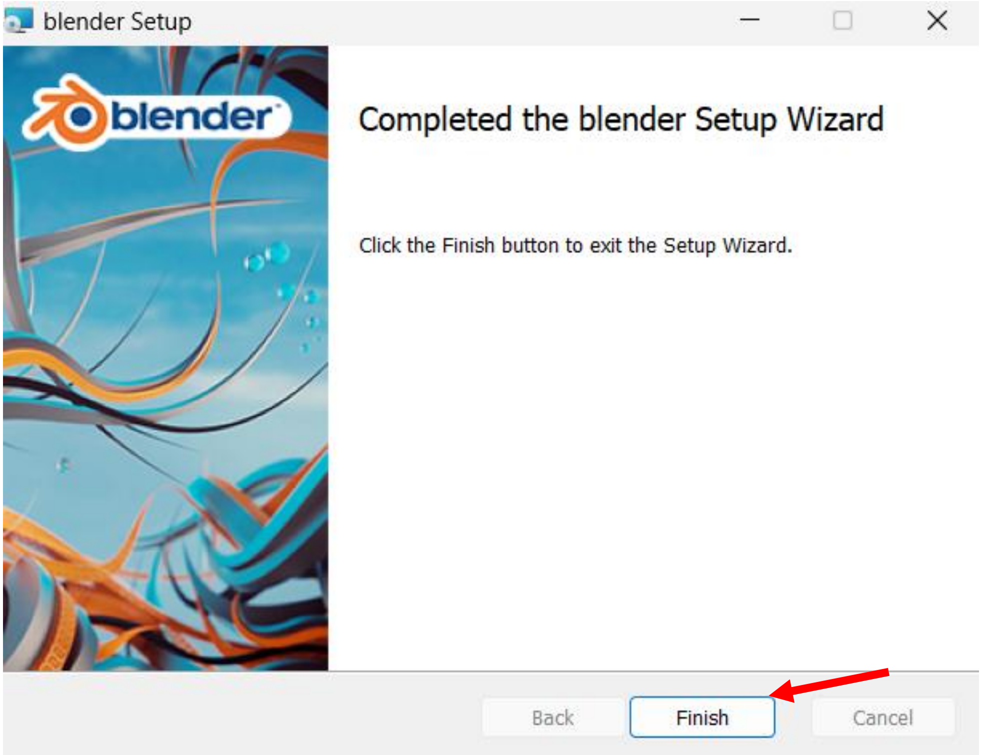


Figure 5.1.3.7 Blender Installation Completed

9. Launch the Blender then click on "General" file type.

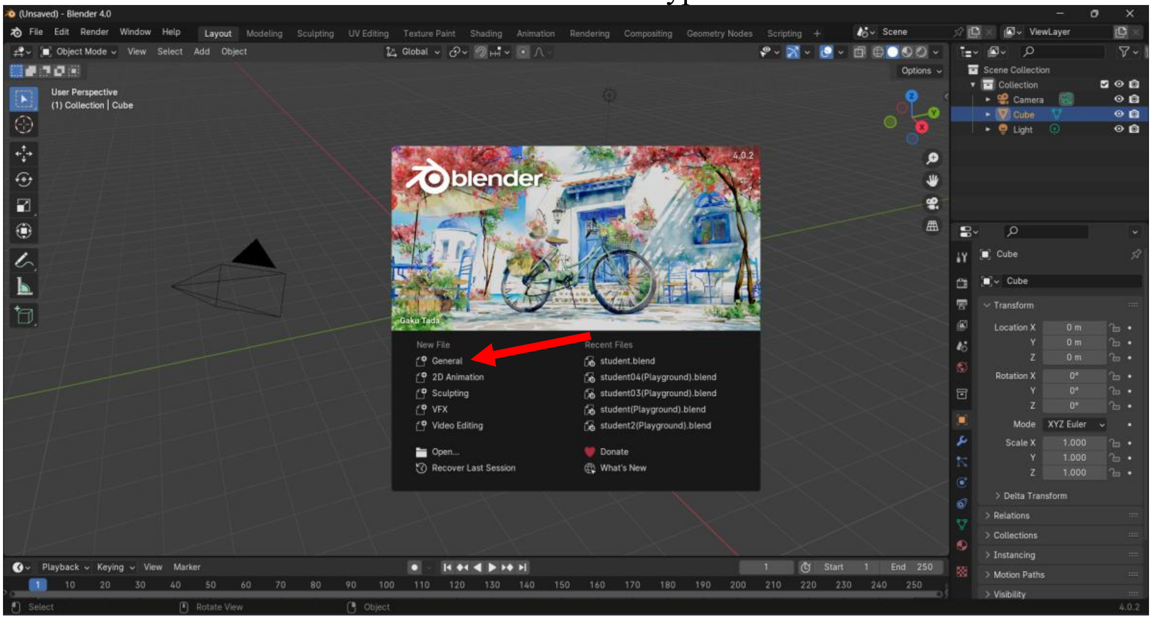


Figure 5.1.3.8 Launch the Blender Project

10. New project will be created.

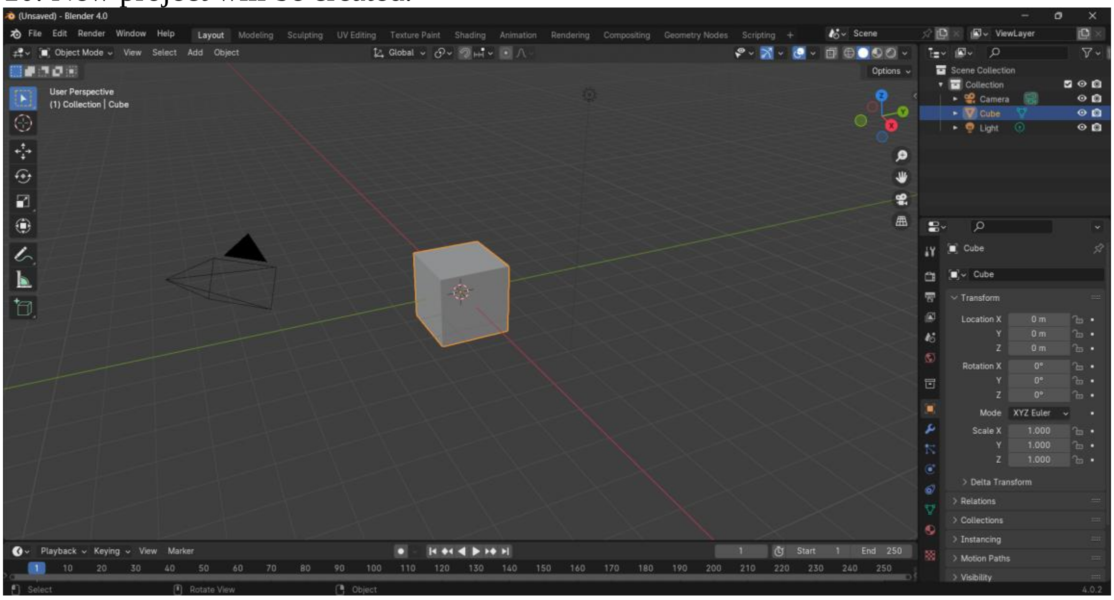


Figure 5.1.3.9 New Blender Project

5.2 Setting and Configuration

5.2.1 Configure Unity to Use Visual Studio

1. Click on “Edit” section at the top left of the Unity Editor.

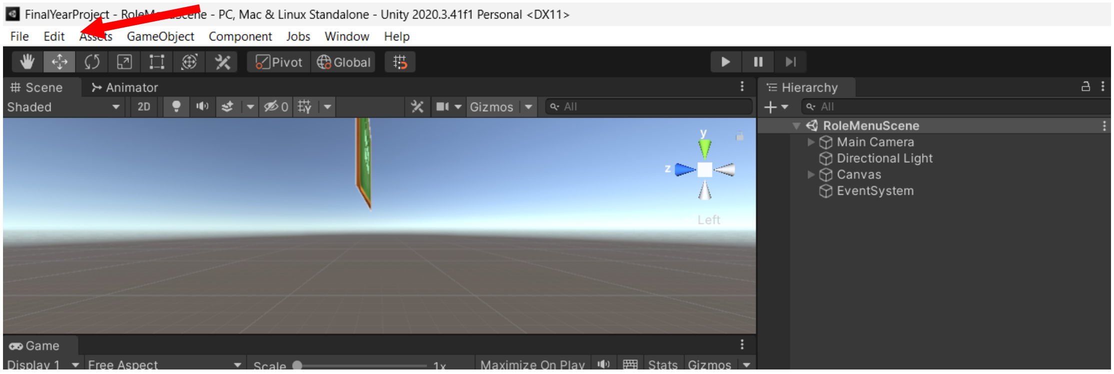


Figure 5.2.1.1 Unity Editor

2. Click on “Preferences” section.

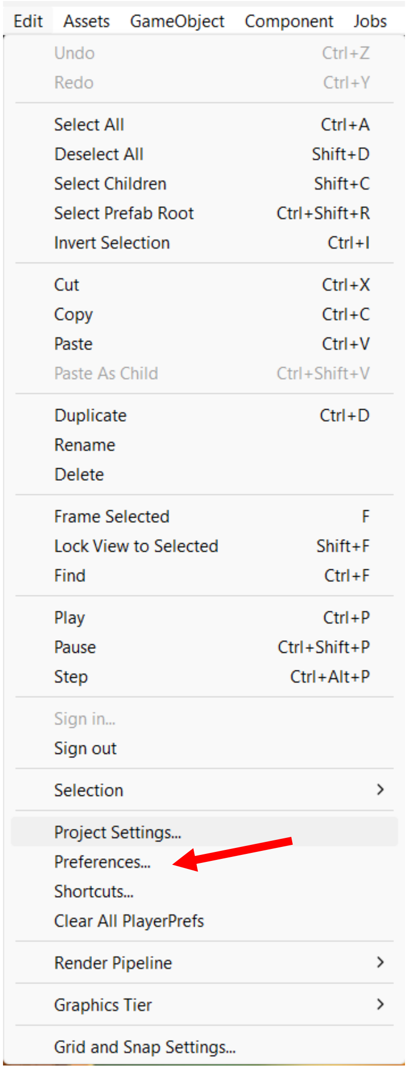


Figure 5.2.1.2 Edit Section

3. Select the “External Script Editor” dropdown list, then choose Visual Studio Community 2022.

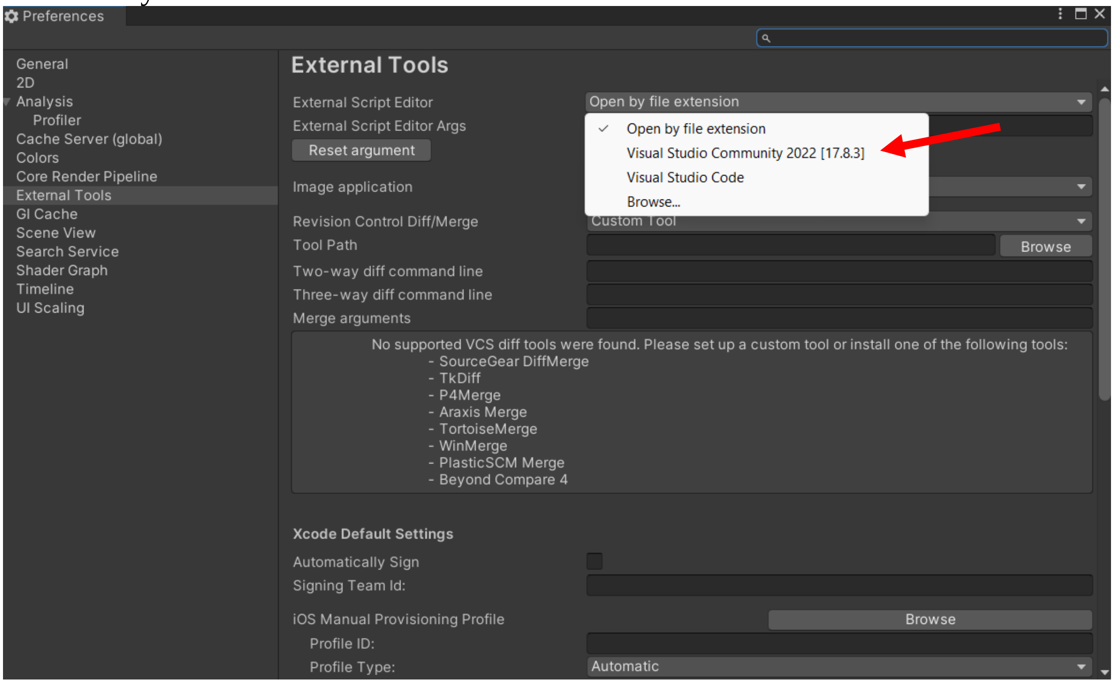


Figure 5.2.1.3 Preferences Section

5.2.2 Installing Packages in Unity

1. Click on “Window” section at top of Unity Editor

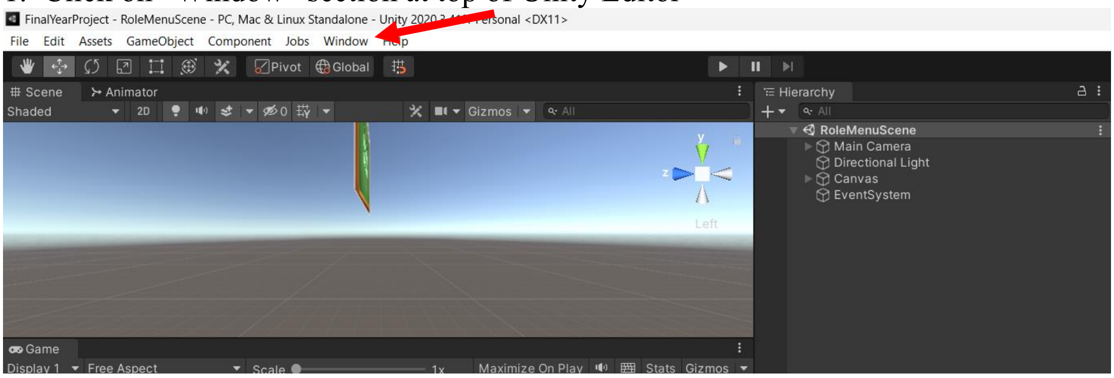


Figure 5.2.2.1 Unity Editor

2. Click on “Package Manager” section.

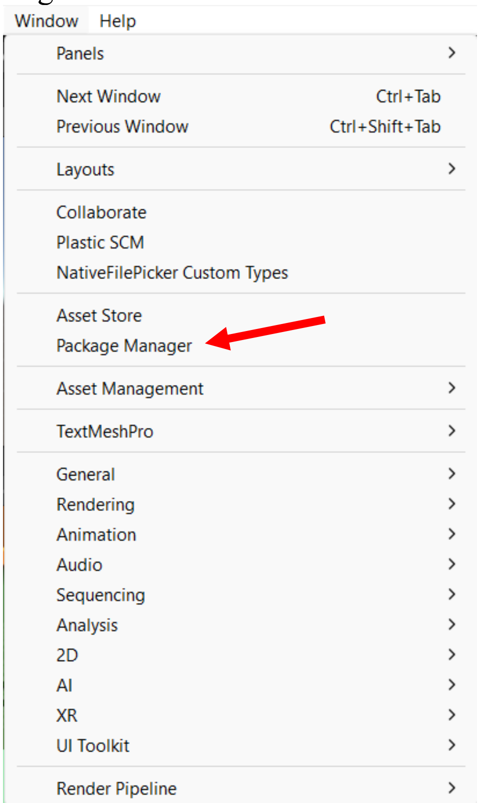


Figure 5.2.2.2 Window Section

3. Click on the search bar at the top right of the Package Manager section, then search “Visual Studio Editor”.

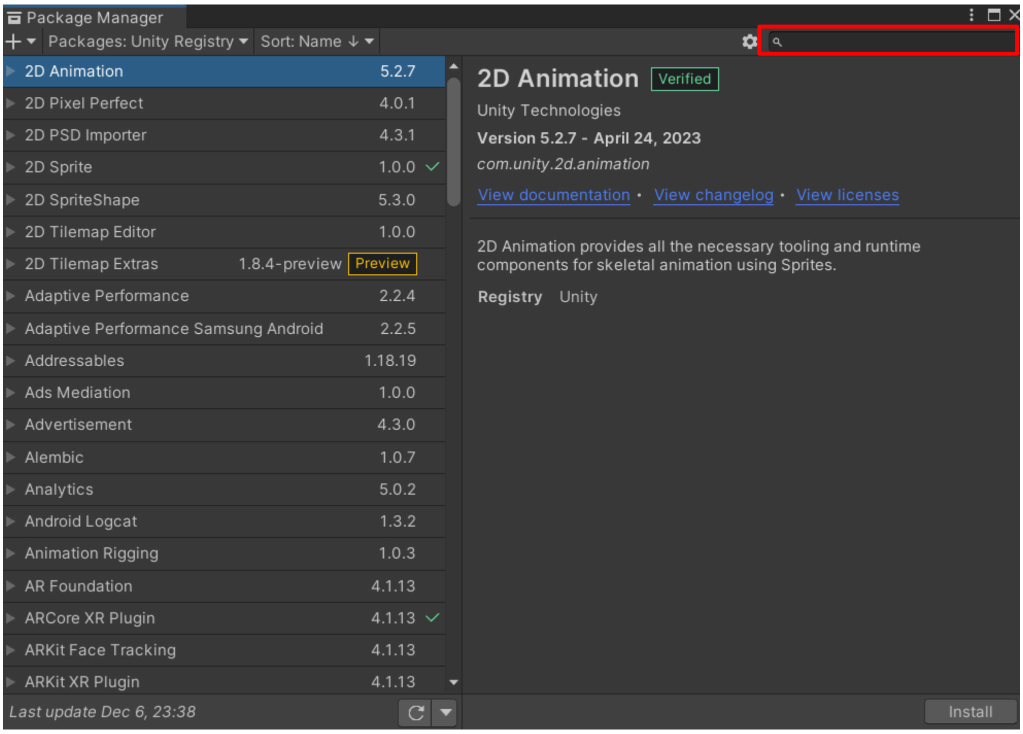


Figure 5.2.2.3 Package Manager Section

4. Click on “Install” button.

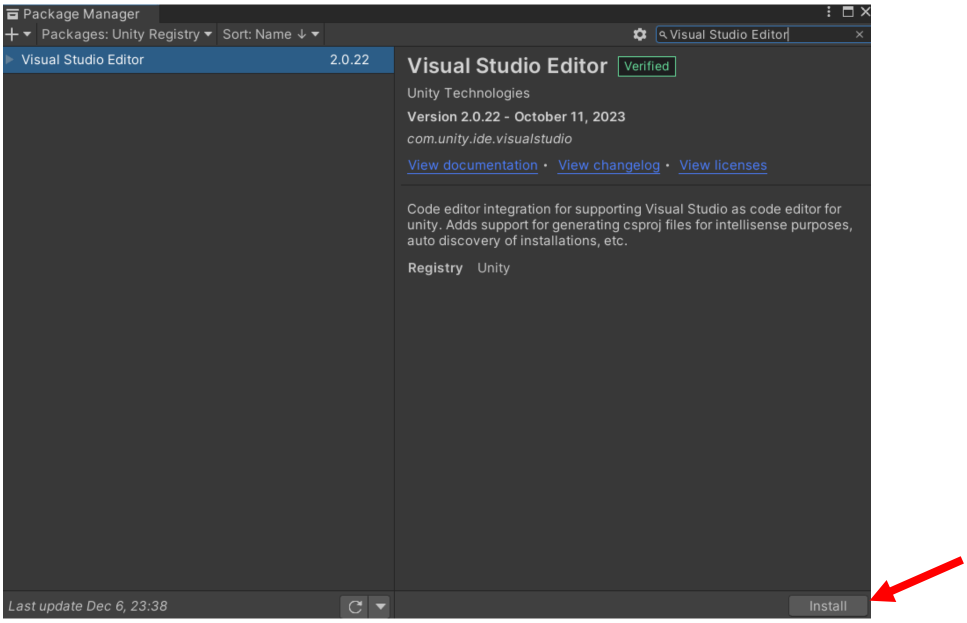


Figure 5.2.2.4 Install Package

5. Repeat Step 1 until Step 4 to install the packages listed in the image below.

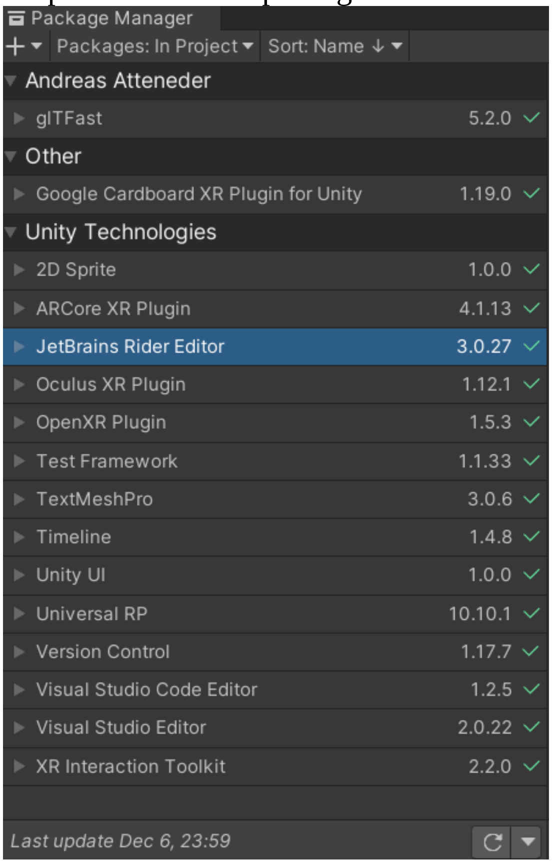


Figure 5.2.2.5 Package List

5.2.3 Enable Cardboard XR Plugin

1. Click on “Edit” section, then select “Project Settings”.

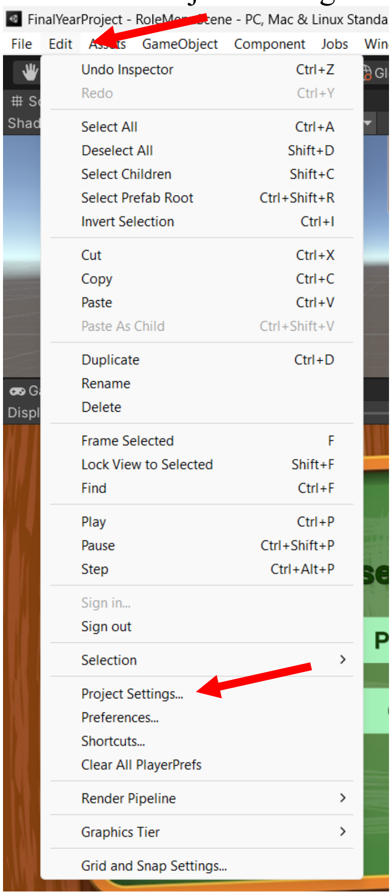


Figure 5.2.3.1 Edit Section

2. Select “XR Plug-in Management” at the left panel.

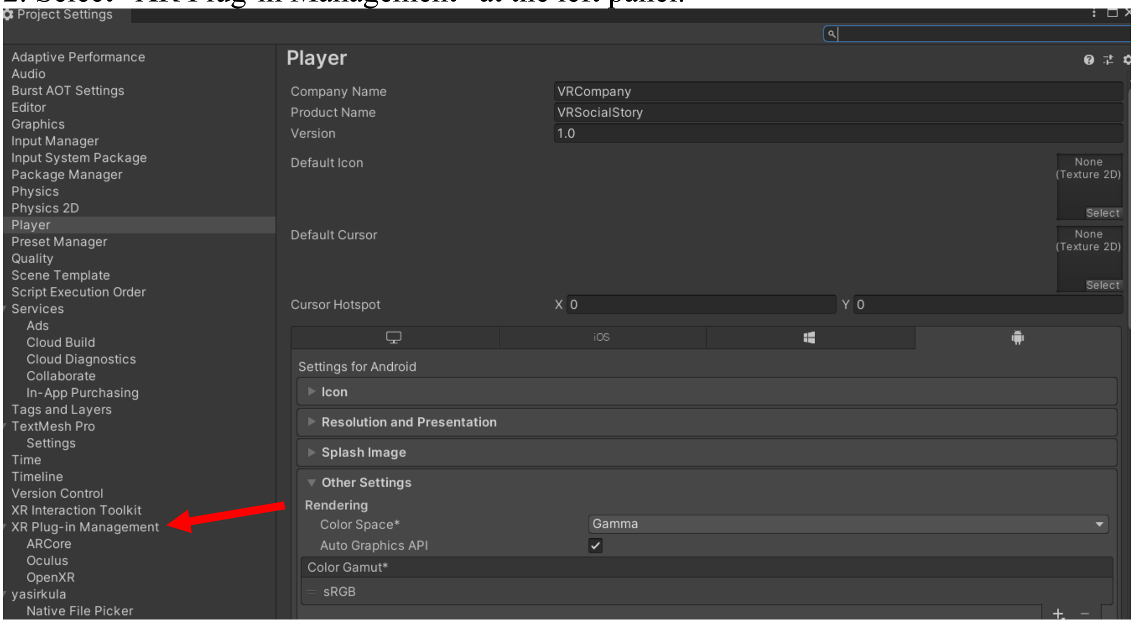


Figure 5.2.3.2 Project Settings

3. Click on Android icon tab, then select “Cardboard XR Plugin” checkbox.

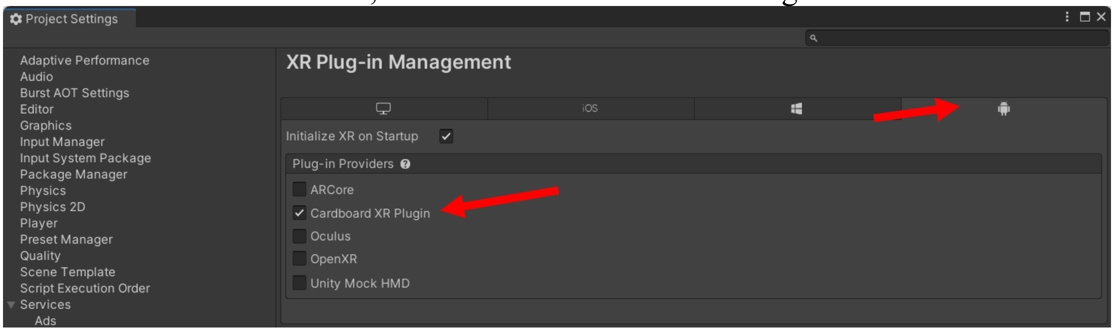


Figure 5.2.3.3 XR Plug-in Management

5.2.4 Configure Player Setting

1. Click on “Edit” section, then select “Project Settings”.

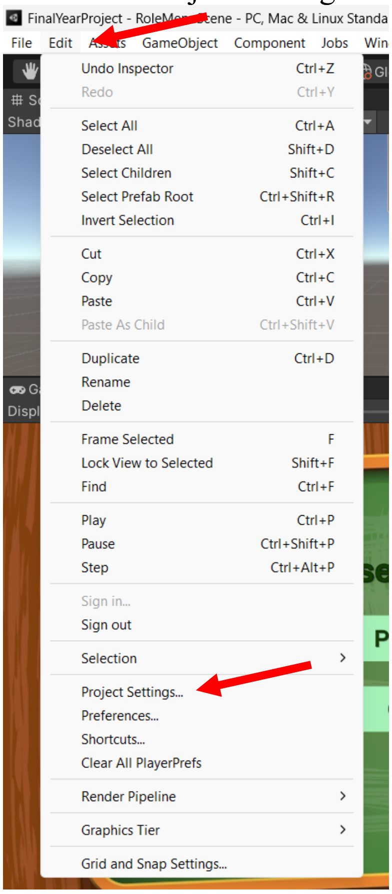


Figure 5.2.4.1 Edit Section

2. Rename “Company Name” and “Product Name” to the name that desired, then change the “Version” to 1.0.

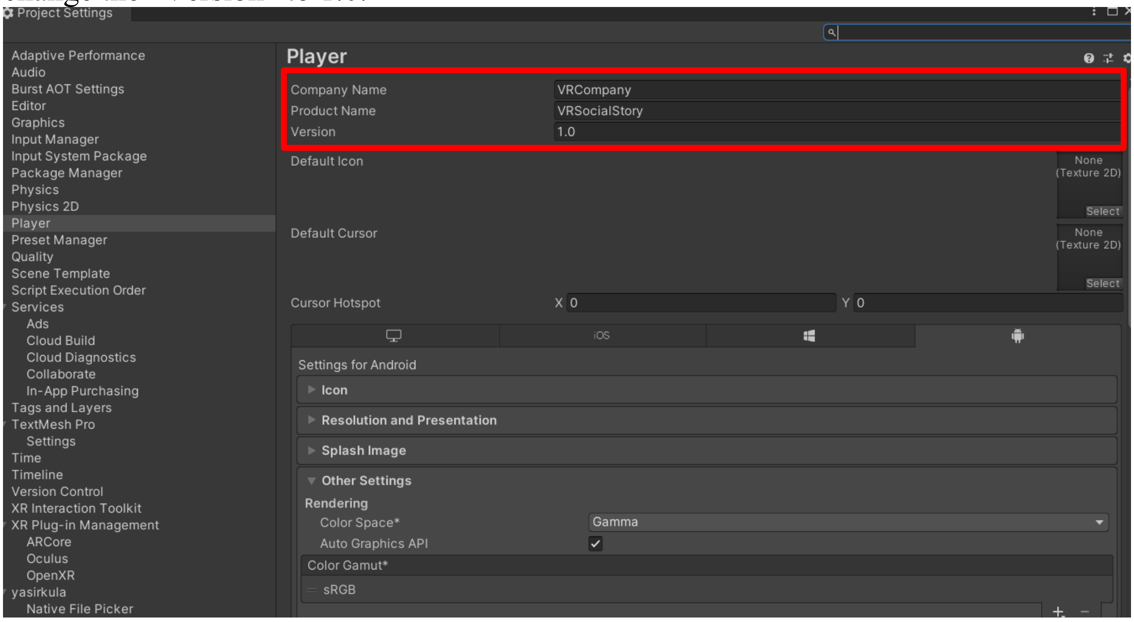


Figure 5.2.4.2 Player Setting

3. Click on Android icon tab, then select “Other Settings” drop down list.

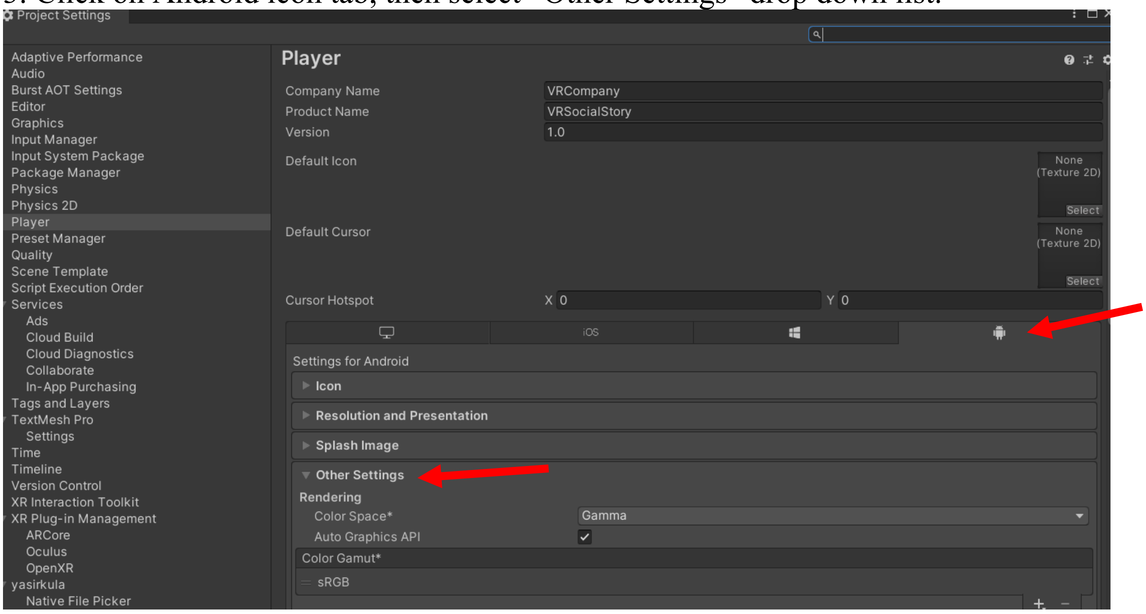


Figure 5.2.4.3 Other Settings

4. Set “Minimum API Level” to Android 7.0, then set “Scripting Backend” to IL2CPP. After that, tick “ARMv7” checkbox.

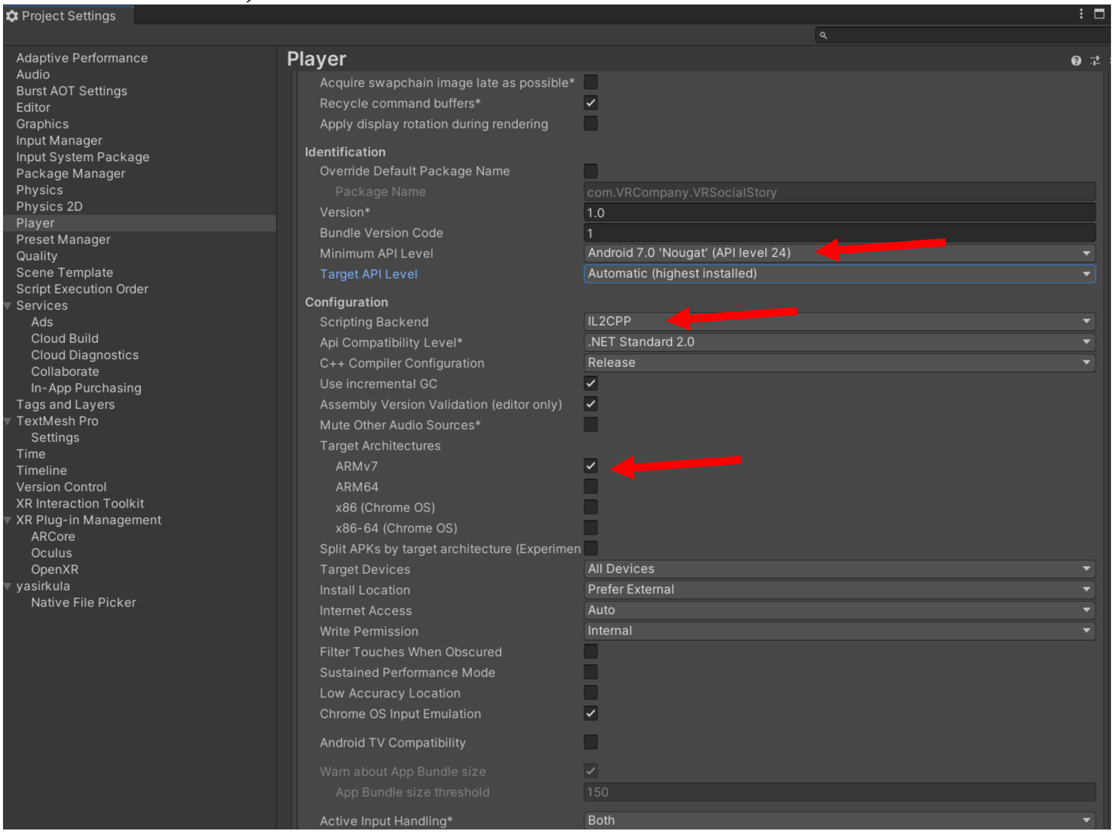


Figure 5.2.4.4 Configure Other Settings

5. Select “Publishing Settings” drop down list.

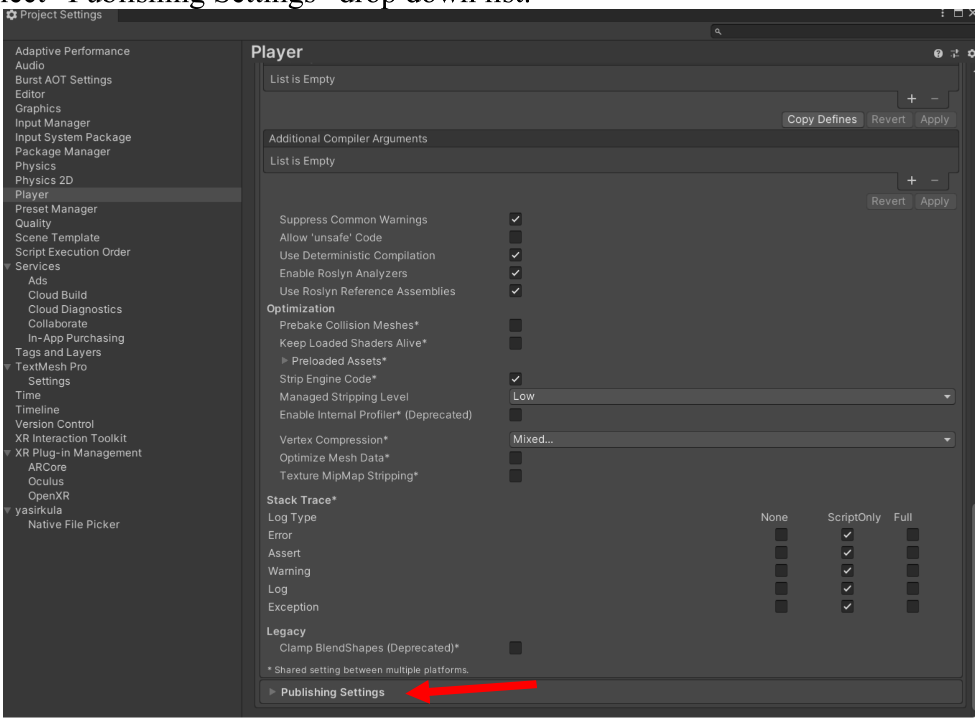


Figure 5.2.4.5 Navigate to Publishing Settings

6. Tick the checkboxes such as “Custom Main Manifest”, “Custom Main Gradle Template”, and “Custom Gradle Properties Template”.

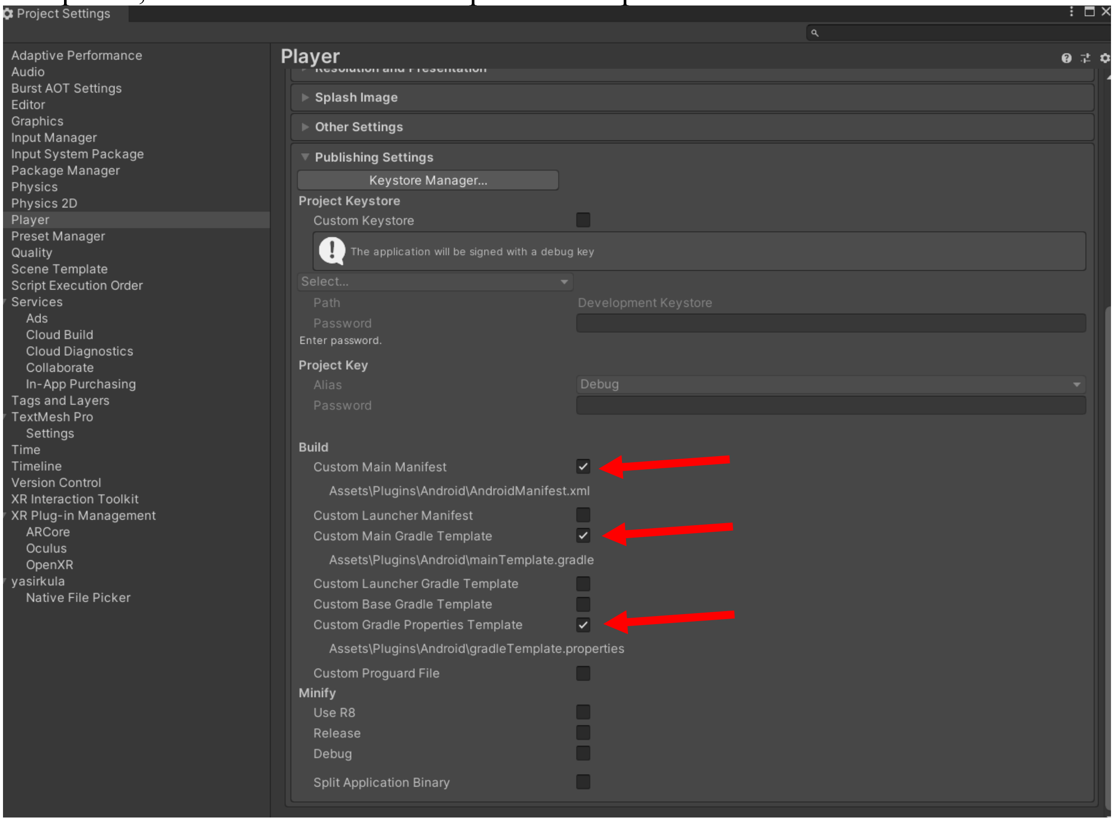


Figure 5.2.4.6 Configure Publishing Settings

5.2.5 Installing Add-ons in Blender

1. Click on “Edit” tab.

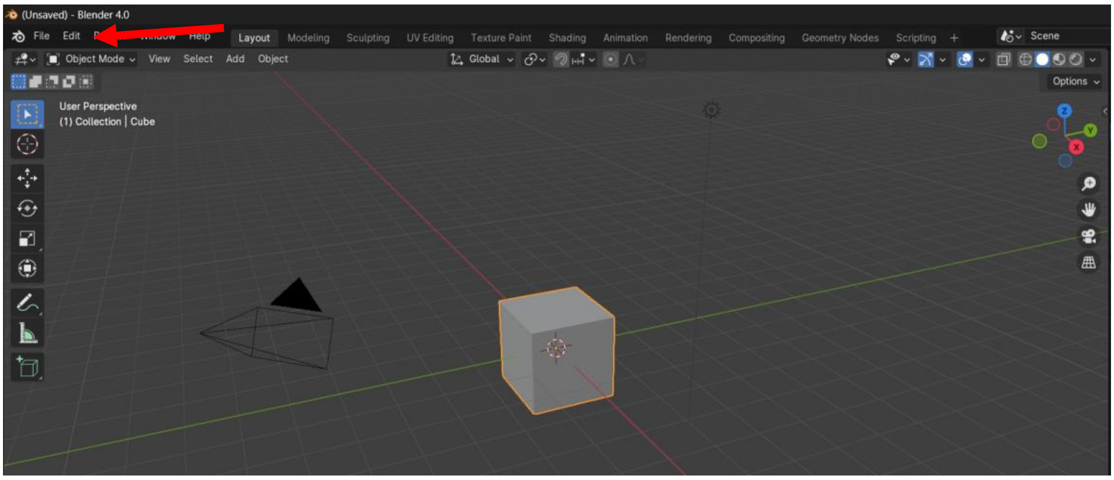


Figure 5.2.5.1 Blender Project

2. Click on “Preferences” option in Edit tab.

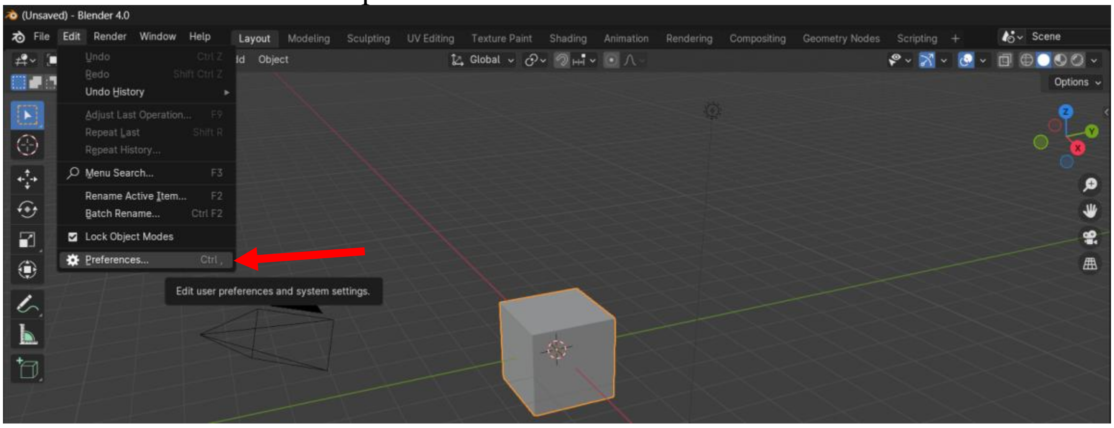


Figure 5.2.5.2 Blender Edit Tab

3. In Add-ons tab, click on search bar then type “Rigify”. Then, click on the checkbox for Rigify.

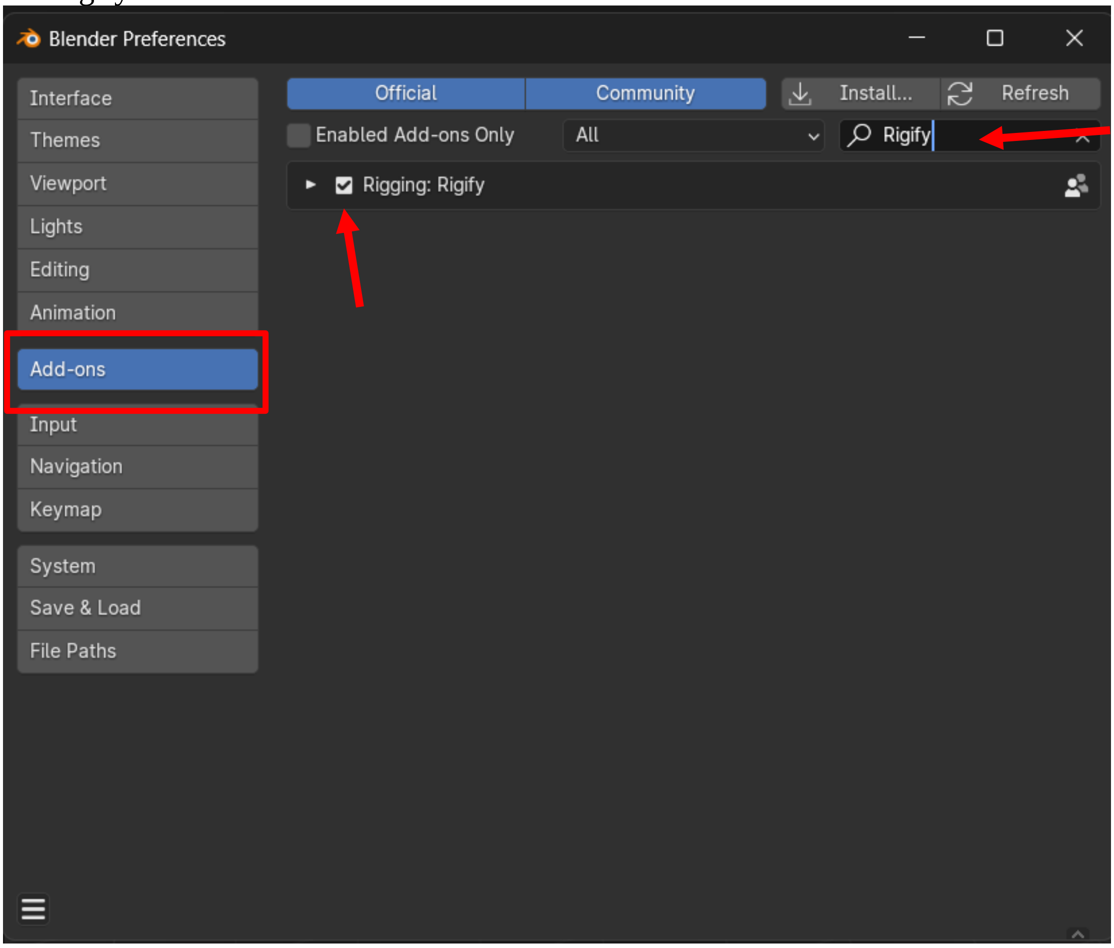


Figure 5.2.5.3 Installing Add-ons in Blender

5.3 System Operation

The images below are the operation of the VR application:

Figure 5.3.1 Main Menu

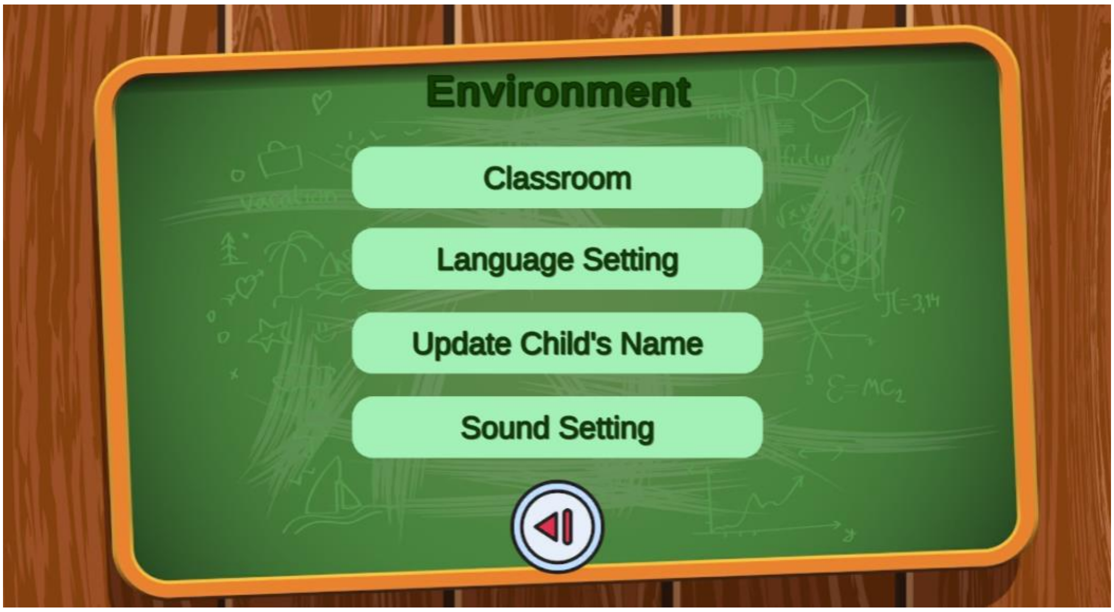


Figure 5.3.2 Environment Menu

Figure 5.3.3 Classroom Customization Menu

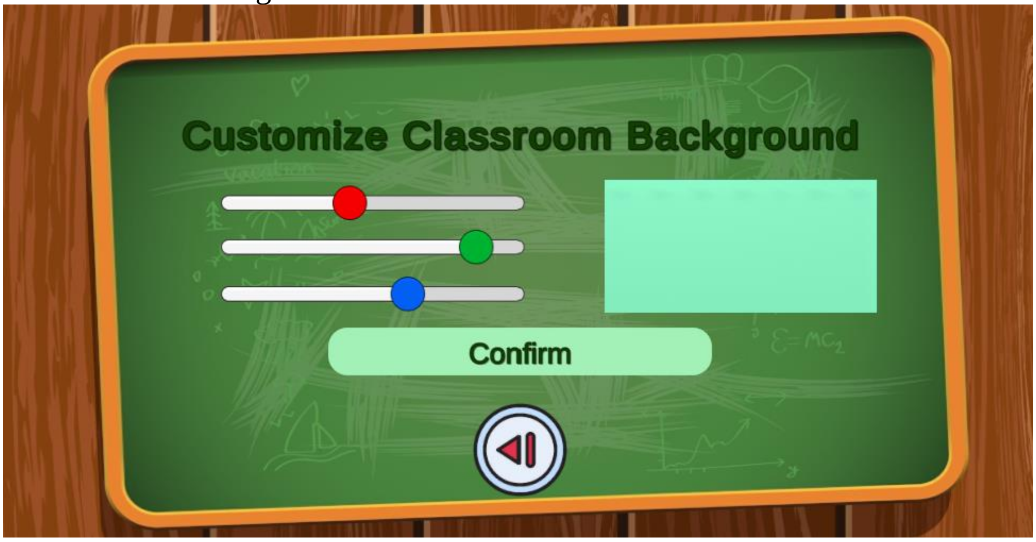


Figure 5.3.4 Modify Classroom Background

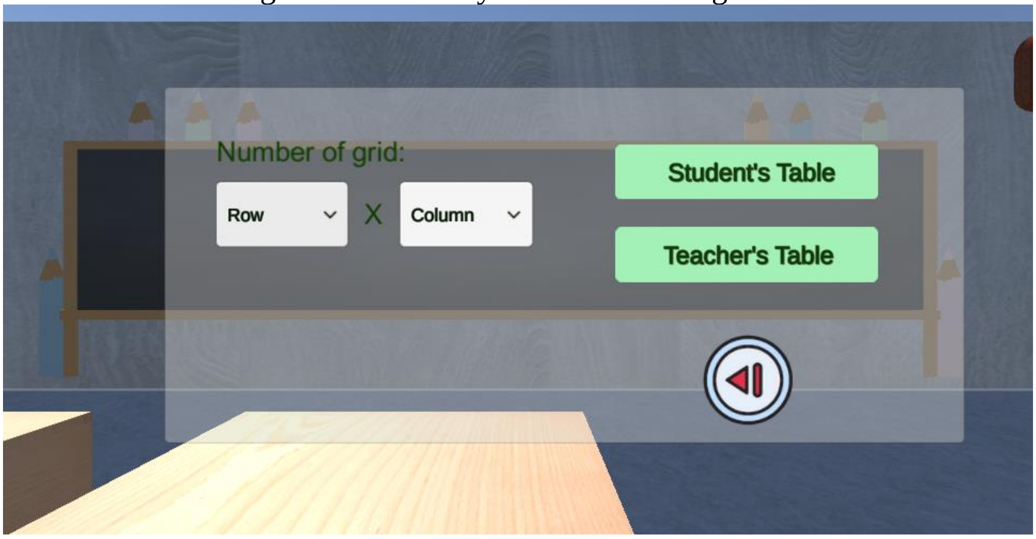


Figure 5.3.5 Modify Classroom Object Interface

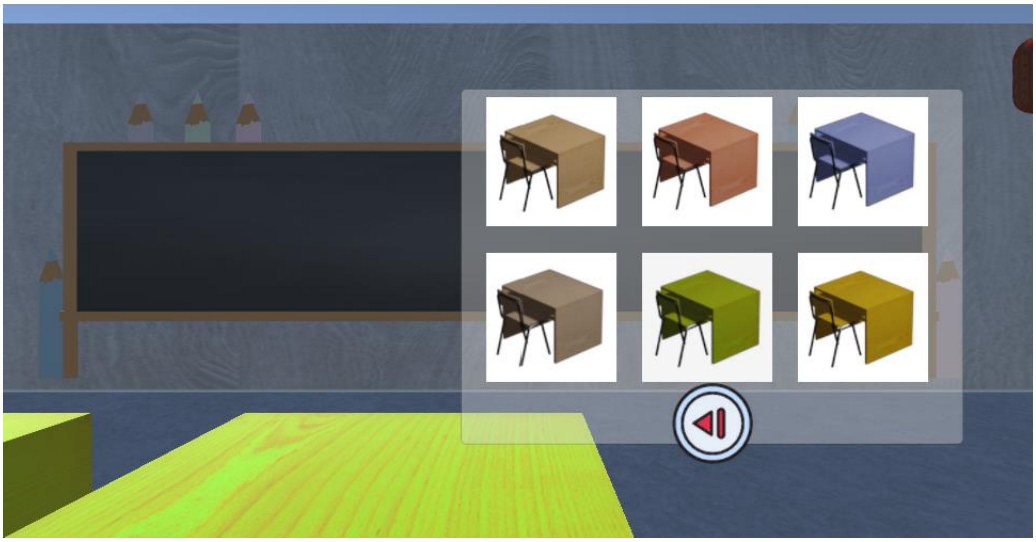


Figure 5.3.6 Student Table Colour Selections

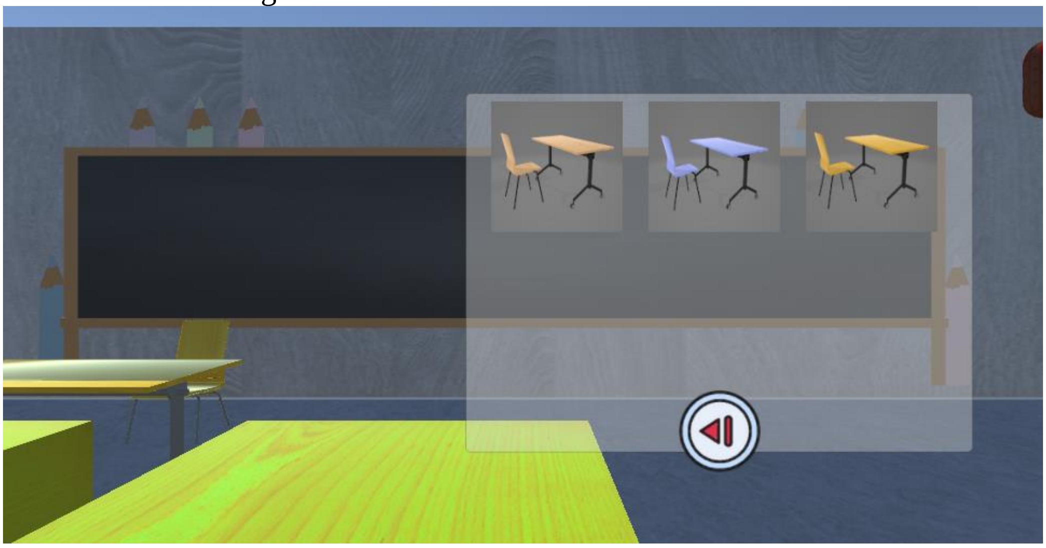


Figure 5.3.7 Teacher Table Colour Selections

Figure 5.3.8 Language Selections

Figure 5.3.9 Update Child's Name

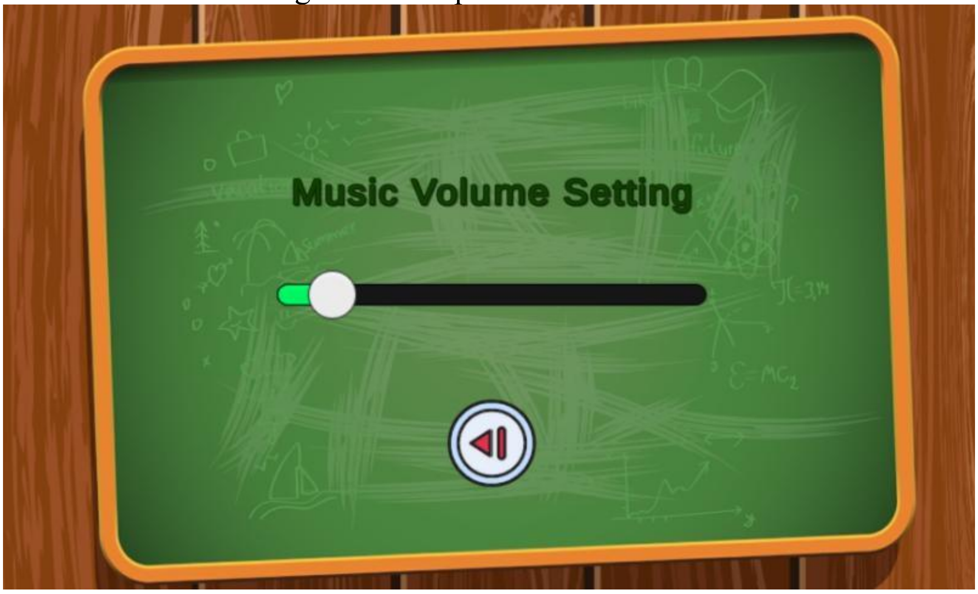


Figure 5.3.10 Sound Setting Menu

Figure 5.3.11 Social Story Selection

Figure 5.3.12 Loading Screen

Figure 5.3.13 User Introduction Scene

Figure 5.3.14 Classmates Introduction Scene

Figure 5.3.15 User Borrows Pencil Scene

Figure 5.3.16 User Getting Laughed by Classmate Scene

Figure 5.3.17 User Getting Touched by Classmate Scene

Figure 5.3.18 Teacher Bring Students to Playground Scene

Figure 5.3.19 User Asking to Play Together Scene

Figure 5.3.20 The Friends Play Tag Scene

Figure 5.3.21 User Play Tag Together Scene

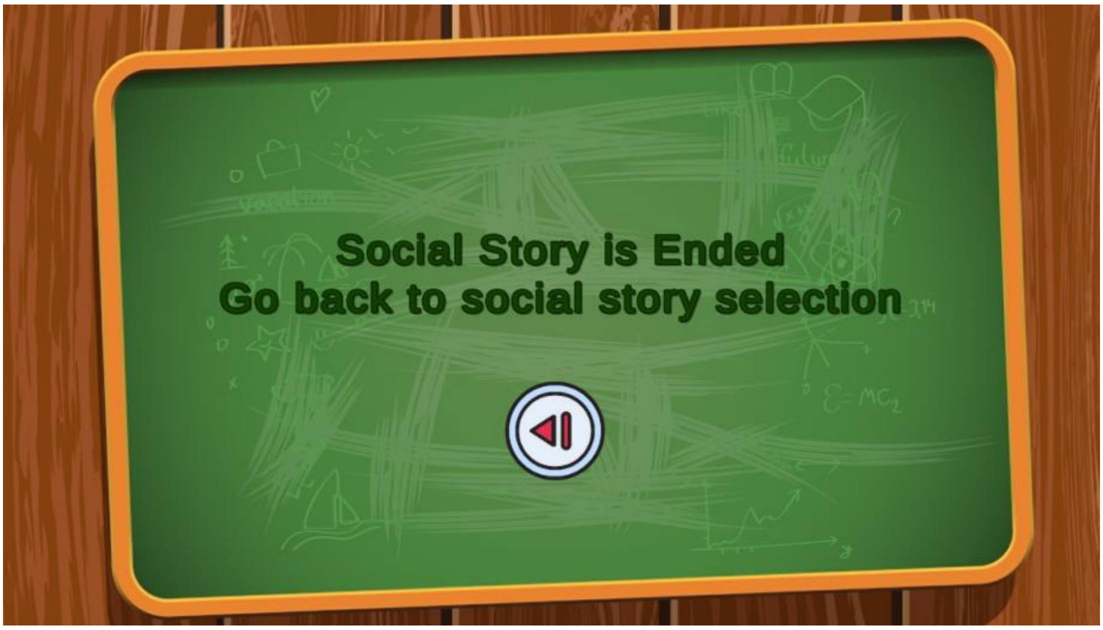


Figure 5.3.22 Social Story End Scene

5.4 Concluding Remark

This chapter will demonstrate the setup of the software used in this project. The settings and configurations for the software will also be discussed. Additionally, this chapter will showcase the operation of the VR application.

CHAPTER 6

System Evaluation and Discussion

6.1 System Testing and Performance Metrics

The VR application allows users to customize the environments and choose the social stories. To ensure a good user experience, a verification plan is done to discover any defect in the VR application.

No	Test Cases	Expected Result	Actual Result	Remark
1	Click on the “Parents” option in main menu.	Direct to environment menu and display options such as classroom, language, and update child’s name.	Direct to environment menu and display options such as classroom, language, and update child’s name.	Pass
2	Click on the “Classroom” option in environment menu.	Direct to classroom setting menu.	Direct to classroom setting menu.	Pass
3	Click on the “Classroom Background” option in classroom setting menu.	Direct to classroom background setting interface.	Direct to classroom background setting interface.	Pass
4	Change the colour of classroom background using the sliders in classroom	The colour of classroom background in social story should be updated.	The colour of classroom background in social story is updated.	Pass

	background setting interface.			
5	Click on back button in classroom background setting interface.	Direct to the classroom setting menu.	Direct to the classroom setting menu.	Pass
6	Click on the “Objects in the Classroom” option in classroom setting menu.	Direct to modify classroom objects interface.	Direct to modify classroom objects interface.	Pass
7	Choose the number of student’s table in modify classroom objects interface.	Spawn the student’s table according to the selected number.	Spawn the student’s table according to the selected number.	Pass
8	Choose the colour of student’s table in modify classroom objects interface.	The student’s table change to selected colour.	The student’s table change to selected colour.	Pass
9	Choose the colour of teacher’s table in modify classroom objects interface.	The teacher’s table change to selected colour.	The teacher’s table change to selected colour.	Pass
10	Click on back button in modify classroom objects interface.	Direct to the classroom setting menu.	Direct to the classroom setting menu.	Pass
11	Click on back button in the classroom setting menu.	Direct to the environment menu.	Direct to the environment menu.	Pass
12	Click on the “Language” option	Direct to the language menu.	Direct to the language menu.	Pass

	in environment menu.			
13	Choose any of the languages (English, Chinese, Malay) in the language menu.	The language of social story should change to the selected language.	The language of social story is changed to the selected language.	Pass
14	Click on back button in language menu.	Direct to the environment menu.	Direct to the environment menu.	Pass
15	Click on the "Sound Setting" option in environment menu.	Direct to the sound setting menu.	Direct to the sound setting menu.	Pass
16	Adjust the background music volume using slider in sound setting menu.	The volume of background music in social story should be updated.	The volume of background music in social story is updated.	Pass
17	Click on back button in sound setting menu.	Direct to the environment menu.	Direct to the environment menu.	Pass
18	Click on update child's name in environment menu.	Direct to the update child's name menu.	Direct to the update child's name menu.	Pass
19	Enter the child's name in the input field then hit enter.	The social story should mention the updated child's name.	The social story mentioned the updated child's name.	Pass
20	Click on back button in update child's name menu.	Direct to the environment menu.	Direct to the environment menu.	Pass
21	Click on back button in environment menu.	Direct to the main menu.	Direct to the main menu.	Pass

22	Click on the “Child ” option in main menu.	Direct to story selection menu.	Direct to story selection menu.	Pass
23	Click on any social story option in story selection menu.	It will launch the selected social story.	The selected social story is launched and it ran without any issue.	Pass
24	Click on back button after the social story is finished	Direct to story selection menu.	Direct to story selection menu.	Pass
25	Click on back button in story selection menu.	Direct to the main menu.	Direct to the main menu.	Pass
26	Click on the “x” button in main menu.	Quite the application.	Quite the application.	Pass

Table 6.1.1 Verification Plan

6.2 Project Challenges

The current project tackles a wide range of challenges during the development of the application, including 3D modelling, animation, audio integration, and virtual reality implementation. These are the project challenges and its solutions:

- **Challenge 1:** Creating avatar models and animations with Blender. Without any prior experience with Blender, learning 3D modelling and rigging from scratch is quite difficult and time-consuming. Designing visually attractive and appealing avatars that effectively portray the desired characters is an artistic activity requiring great attention to detail and an in-depth understanding of 3D modelling techniques. Moreover, the animation process itself presents its own set of challenges. A good understanding of body mechanics, timing, and acting concepts are essential for animating avatars in a way that produces realistic and natural motions. Ensuring that the animations coordinate flawlessly with the social stories is critical for maintaining immersion and consistency in the virtual environment. This synchronization operation might be very difficult since it requires exact timing and a careful attention to detail. Another layer of complication to the animation concerns is keeping the avatar's running animation in sync with its position within the social story.

- **Solution 1:** To overcome the lack of prior experience with Blender for 3D modelling and animation, a significant amount of time was dedicated to self-learning these skills from the ground up. Online tutorials, documentation, and forums were extensively utilized to understand the fundamentals of 3D modelling techniques such as polygonal modelling, texturing, and others. Character modelling was prioritized in order to build visually appealing avatar models that effectively represented the desired characters in the social story. A thorough grasp of concepts like time, anticipation, spacing, and body mechanics was necessary for animations. The online resources were extremely useful for understanding the Blender animation workflow, rigging principles like as skinning and weight painting, and acting approaches for creating realistic and natural character motions. To preserve immersion and consistency inside the virtual environment, the avatar animations were carefully timed to match the language and happenings of the social story sequences. A blend tree was utilized for blending the running animation with its position smoothly based on the avatar's speed.

- **Challenge 2:** The integration of audio voiceovers and background music. It takes effort to find the right tools and learn how to utilize them successfully for audio voiceover creation. Maintaining a seamless and engaging experience requires that the audio voiceovers and the social story dialogues be in sync. In addition, the project requires audio voiceovers in a variety of languages, including Chinese, English, and Malay, which adds to the burden and complexity. In terms of the background music, it is difficult to make sure that it keeps playing when the social story scenes change without any disruptions. Also, there is a need to discover an alternative way for the background music to play in the sound settings menu, allowing parents to listen to the music while choose their preferred volume.

- **Solution 2:** Numerous audio editing tools like Audacity and text-to-speech software were researched. Microsoft Clipchamp was chosen as the audio editing tool and text-to-speech software. The usage of Microsoft Clipchamp was learned through documentation and tutorials. Microsoft Clipchamp allows the production of clear, high-quality audio voiceovers in English, Chinese, and Malay to meet a variety of linguistic needs. Precise timings of the voiceovers were measured to ensure seamless coordination with the social story dialog sequences. The voiceovers were divided into audio clips that played at appropriate points in the social story. Within Unity, audio source components were programmed to start or stop voiceovers and background music at designated times. In the sound setting menu, the background music audio source was set to play and loop, allowing parents to listen to the background music and adjust volume before entering the social story. The background music would pause after exit the sound setting menu, the background music.

- **Challenge 3:** The creation of the VR mode script. This script is responsible for activating and disabling the VR mode at various scenes of the application. With the end of updates for the Google Cardboard VR framework, which is now considered a legacy technology, locating relevant materials and documentation online has become challenging. Despite the lack of resources and support, a substantial amount of time to research and brainstorming are required in order to finish the VR mode script.

- **Solution 3:** As the Google Cardboard VR framework is no longer actively updated, locating existing documentation and support resources online proved incredibly difficult. This involved a significant amount of work spent reviewing existing documentation, outdated forums and comments, and other sites that discussed Cardboard SDK usage. The VR mode script was gradually developed via careful analysis of available examples and experimentation. The VR mode script was developed to automate tasks such as turning on or off VR mode programmatically and querying VR capabilities at runtime. The difficulties in incorporating VR mode into Unity across several scenes while maintaining the desired application state were solved by careful scripting, testing, and incremental improvement.

6.3 Objectives Evaluation

The primary objective of developing an immersive, accessible, and customizable VR application with the Unity engine and Google Cardboard VR has been accomplished. The developed VR application offers engaging and repeatable experiences for primary school students with ASD to practice essential social skills and handle difficult scenarios linked to forming friendships in the social stories. The virtual environment has been carefully adjusted to each student's specific requirements and preferences, resulting in an engaging and inclusive learning experience.

One of the key sub-objectives, developing a VR application that allows parents to customize various aspects of the VR environment, language settings, and social story content, has been effectively accomplished. The application has a straightforward and user-friendly interface that allows parents to adjust numerous features of the virtual environment. Parents may tailor the classroom environment, change the selection of language, adjust sound setting and even update their child’s name into social stories, which can increase their child's enthusiasm and engagement.

Furthermore, the third objective has been successfully achieved by creating meaningful and relevant social stories that reflect the specific experiences and obstacles that autistic primary school students encounter. By getting a thorough knowledge of the target audience's needs and interests, this project has created social stories that address specific areas of difficulty, such as forming friendships, sharing, taking turns, and active listening. These personalized stories, based on real-life circumstances, seek to convey useful lessons and contribute to autistic children' overall social development journeys by encouraging interaction and teaching crucial social skills.

6.4 Concluding Remark

This chapter shows the verification plan to discover any defect in the VR application. Moreover, this chapter will discuss the challenges faced during the development of the application and the evaluation of the project‘s objectives.

CHAPTER 7

Conclusion and Recommendation

7.1 Conclusion

In conclusion, the development of this customizable VR application met its goals of providing an immersive, accessible, and tailored learning environment for autistic primary school students. By harnessing the amazing features of the Unity engine and Google Cardboard VR, the project has created an entertaining and repeatable platform for students to practice critical social skills and handle difficult scenarios related to forming friendships in a simulated school setting.

One of the key strengths of this project lies in its emphasis on customization and tailored experiences. With its user-friendly interface, the application gives parents the ability to customize the VR environment, language settings, adjust sound setting and update child’s name into social story to suit their child's individual requirements and preferences. This customization promotes motivation, inclusion, and relevance to each student's specific needs and learning objectives.

Additionally, the project has excelled at developing meaningful and relevant social stories that are consistent with the real-life experiences and struggles of autistic primary school students. These personalized stories, based on real-life circumstances, seek to impart essential lessons and contribute to autistic students’ overall social development journeys by encouraging interaction and teaching crucial social skills in an immersive and secure virtual environment.

The successful completion of this project not only meets its objectives, but also sets the way for future research and developments in harnessing VR technology for customized and effective interventions in autism education and therapy. By showcasing the possibility of customizable VR experiences, this study adds to a better understanding of how technology may be used to improve social learning outcomes for people with a variety of needs and skills on the autism spectrum.

7.2 Recommendation

Although the current customizable VR application has accomplished its goals and made great progress toward offering autistic primary school students an immersive and customized learning environment, there are a few recommendations that could improve its effectiveness and broaden its impact.

First, it is recommended that powerful artificial intelligence (AI) and machine learning techniques be integrated to allow for real-time adaptation and customization of social stories and virtual settings. Using AI algorithms, the application may dynamically modify difficulty levels, situations, and challenges based on each student's success, learning habits, and areas of strength or weakness. This would provide a more personalized and responsive learning experience, increasing engagement and skill development.

To further enhance the immersive and personalized experience offered by the VR application, it is highly recommended that future development efforts focus on incorporating robust customization capabilities within the playground scene. This would include adding features that allow parents to customize the playground's layout, equipment placement, surface textures, and colours, as well as populating the area with configurable non-playable characters (NPCs) representing the student's classmates or friends. By enabling the modification of the playground environment to accurately recreate each student’s real-world setting, including the ability to customize NPC looks and actions, it may generate a deeper feeling of familiarity, comfort, and engagement.

Another recommendation is to broaden the application's accessibility capabilities and explore alternate input methods. Incorporating voice instructions, eye tracking, or brain-computer interfaces might make the VR experience more accessible for students who have different physical or cognitive capacities, guaranteeing that no one falls behind in this revolutionary learning method.

In addition, it is suggested to think about implementing multiplayer or collaborative aspects into the VR experience. This would allow autistic students to practice social interactions with real-life classmates or teachers, as well as virtual characters. Such collaborative activities may promote cooperation, communication skills, and a feeling of community, so improving social learning outcomes.

Additionally, the project could benefit from establishing partnerships and collaborations with educational institutions, autism research centres, and support organizations. These collaborations may enable the interchange of information, skills, and resources, resulting in more comprehensive and evidence-based methods to leveraging VR technology for autism therapies.

Finally, continuous user feedback and evaluation should be prioritized to ensure that the application remains relevant and effective. Regularly receiving feedback from autistic students, parents, and therapists would discover areas for improvement, address growing needs, and adapt the application to changing best practices in autism support and education.

REFERENCES

[1] N. Didehbani, T. Allen, M. Kandalaft, D. Krawczyk, and S. Chapman, "Virtual reality social cognition training for children with high functioning autism," Computers in Human Behavior, vol. 62, pp. 703–711, Sep. 2016. doi:10.1016/j.chb.2016.04.033.

[2] Unity Technologies, "Build VR experiences with unity," Unity, https://unity.com/solutions/vr#accelerate-solutions (accessed Nov. 25, 2023).

[3] "Nanite virtualized geometry," in Unreal Engine | Unreal Engine 5.3 Documentation, https://docs.unrealengine.com/5.3/en-US/nanite-virtualized-geometry-in-unreal-engine/ (accessed Nov. 25, 2023).

[4] "Lumen technical details," in Unreal Engine | Unreal Engine 5.0 Documentation, https://docs.unrealengine.com/5.0/en-US/lumen-technical-details-in-unreal-engine/ (accessed Nov. 25, 2023).

[5] "VR template," in Unreal Engine | Unreal Engine 5.0 Documentation, https://docs.unrealengine.com/5.0/en-US/vr-template-in-unreal-engine/ (accessed Nov. 25, 2023).

[6] "Meta quest 3 coming this Fall + lower prices for Quest 2," Meta, https://about.fb.com/news/2023/06/meta-quest-3-coming-this-fall/ (accessed Nov. 25, 2023).

[7] "Headset," Valve Index® - Upgrade your experience - Valve Corporation, https://www.valvesoftware.com/en/index/headset (accessed Nov. 26, 2023).

[8] S. N. V. Yuan and H. H. S. Ip, "Using virtual reality to train emotional and social skills in children with autism spectrum disorder," London J Prim Care (Abingdon), vol. 10, no. 4, pp. 110–112, Jul. 2018, doi: 10.1080/17571472.2018.1483000.

[9] M. Alimanova, A. Soltiyeva, M. Urmanov, and S. Adilkhan, "Developing an Immersive Virtual Reality Training System to Enrich Social Interaction and Communication Skills for

Children With Autism Spectrum Disorder,” in SIST 2022 - 2022 International Conference on Smart Information Systems and Technologies, Proceedings, Institute of Electrical and Electronics Engineers Inc., 2022. doi: 10.1109/SIST54437.2022.9945733.

[10] Universitas Negeri Malang. Fakultas Ilmu Pendidikan and Institute of Electrical and Electronics Engineers, 2020 6th ICET, International Conference on Education and Technology : October 17, 2020, Malang, East Java, Indonesia.

[11] Institute of Electrical and Electronics Engineers, 2019 International Biomedical Instrumentation and Technology Conference (IBITeC).

[12] B. Lutkevich and S. Lewis, “What is the waterfall model? - definition and guide,” Software Quality, https://www.techtarget.com/searchsoftwarequality/definition/waterfall-model (accessed Dec. 1, 2023).

[13] “Incremental model (software engineering) - javatpoint,” www.javatpoint.com, https://www.javatpoint.com/software-engineering-incremental-model (accessed Dec. 2, 2023).

[14] Person, “Rapid application development (RAD): Definition, Steps & Full Guide,” Kissflow, https://kissflow.com/application-development/rad/rapid-application-development/ (accessed Dec. 3, 2023).

[15] “Agile model (software engineering) - javatpoint,” www.javatpoint.com, https://www.javatpoint.com/software-engineering-agile-model (accessed Dec. 4, 2023).

FINAL YEAR PROJECT WEEKLY REPORT

(Project II)

Trimester, Year: 3, 3	**Study week no.:2**
Student Name & ID: Tee Zi Jun & 20ACB00978	
Supervisor: Ts Dr Goh Hock Guan	
Project Title: Customizable VR Design of Friends Interaction for Social Story	

1. WORK DONE

[Please write the details of the work done in the last fortnight.]

- Review the project and plan what tasks to be done.
- New social story map is created.

2. WORK TO BE DONE

- Add audio elements into the social story.
- Create new social story.
- Add sound setting.

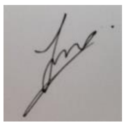
3. PROBLEMS ENCOUNTERED

- No problem encountered so far.

4. SELF EVALUATION OF THE PROGRESS

- Slightly behind the progress.

Supervisor's signature



Student's signature

FINAL YEAR PROJECT WEEKLY REPORT

(Project II)

Trimester, Year: 3, 3	**Study week no.:4**
Student Name & ID: Tee Zi Jun & 20ACB00978	
Supervisor: Ts Dr Goh Hock Guan	
Project Title: Customizable VR Design of Friends Interaction for Social Story	

1. WORK DONE

[Please write the details of the work done in the last fortnight.]

- Add background music.
- Complete the new social story dialogues and 70% of the animations.

2. WORK TO BE DONE

- Add audio voiceovers.
- Complete the new social story.
- Complete the sound setting.

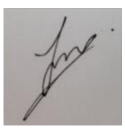
3. PROBLEMS ENCOUNTERED

- The background music will cut off while switching the social story scenes.

4. SELF EVALUATION OF THE PROGRESS

- Progress on track, put more effort on development.

Supervisor's signature



Student's signature

FINAL YEAR PROJECT WEEKLY REPORT

(Project II)

Trimester, Year: 3, 3	**Study week no.:6**
Student Name & ID: Tee Zi Jun & 20ACB00978	
Supervisor: Ts Dr Goh Hock Guan	
Project Title: Customizable VR Design of Friends Interaction for Social Story	

1. WORK DONE

[Please write the details of the work done in the last fortnight.]

- Complete the GUI for sound setting.
- Fix background music issue.
- Complete the new social story but got some issues.

2. WORK TO BE DONE

- Complete the audio voiceovers.
- Complete the sound setting.
- Complete the new social story.

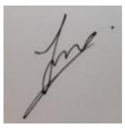
3. PROBLEMS ENCOUNTERED

- User can adjust the volume of background music while listen to it in sound setting. After exit the sound setting, it will end the background music. It will not play again in social story.
- Need to figure out how to sync the running animation with its position while running.

4. SELF EVALUATION OF THE PROGRESS

- Slightly behind the track due to the issues.

Supervisor's signature



Student's signature

FINAL YEAR PROJECT WEEKLY REPORT

(Project II)

Trimester, Year: 3, 3	**Study week no.:8**
Student Name & ID: Tee Zi Jun & 20ACB00978	
Supervisor: Ts Dr Goh Hock Guan	
Project Title: Customizable VR Design of Friends Interaction for Social Story	

1. WORK DONE

[Please write the details of the work done in the last fortnight.]

- Complete the new social story.
- Complete the sound setting.
- Integrate audio voiceover into social story.

2. WORK TO BE DONE

- Install the application in an Android device.
- Build the VR mode script.

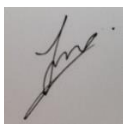
3. PROBLEMS ENCOUNTERED

- No problems so far.

4. SELF EVALUATION OF THE PROGRESS

- Progress is on track.

Supervisor's signature



Student's signature

FINAL YEAR PROJECT WEEKLY REPORT

(Project II)

Trimester, Year: 3, 3	**Study week no.:10**
Student Name & ID: Tee Zi Jun & 20ACB00978	
Supervisor: Ts Dr Goh Hock Guan	
Project Title: Customizable VR Design of Friends Interaction for Social Story	

1. WORK DONE

[Please write the details of the work done in the last fortnight.]

- Complete the VR mode script.
- Install the application into Android device.

2. WORK TO BE DONE

- Finish FYP2 report.

3. PROBLEMS ENCOUNTERED

- No problems so far.

4. SELF EVALUATION OF THE PROGRESS

- Progress is on track.

Supervisor's signature

Student's signature

FINAL YEAR PROJECT WEEKLY REPORT

(Project II)

Trimester, Year: 3, 3	**Study week no.:12**
Student Name & ID: Tee Zi Jun & 20ACB00978	
Supervisor: Ts Dr Goh Hock Guan	
Project Title: Customizable VR Design of Friends Interaction for Social Story	

1. WORK DONE

[Please write the details of the work done in the last fortnight.]

- Complete 80% of the FYP2 report.

2. WORK TO BE DONE

- Complete FYP and prepare for presentation.

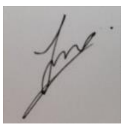
3. PROBLEMS ENCOUNTERED

- No problem so far.

4. SELF EVALUATION OF THE PROGRESS

- Progress on track and able to submit the report on time.

Supervisor's signature



Student's signature

POSTER

CUSTOMIZABLE VR DESIGN OF FRIENDS INTERACTION FOR SOCIAL STORY
UTAR
UNIVERSITI TUNKU ABDUL RAHMAN
INTRODUCTION
This project is to develop a customizable VR application for primary students with Autism Spectrum Disorder (ASD) to rehearse social interactions involved in forming new friendships. Parents can tailor backgrounds and scenarios to match with their child's preferences.
OBJECTIVES
• To create an immersive, accessible, and customized VR application.
• Allows parents to customize the VR environment, language, and social stories.
• To create social stories that are relevant .
SOCIAL STORY
• Making new friends in classroom.
• Playing at the playground.
This is our classmate, Joseph.
We're playing tag. It's really fun, you just have to run and try not to get caught.
CUSTOMIZATION FEATURES
• Customize classroom background colour and objects.
• Choose the languages such as English, Chinese, and Malay.
• Adjust the sound setting.
• Update the child's name into social story.
Prepared by Tee Zi Jun
Supervised by Ts Dr Goh Hock Guan

PLAGIARISM CHECK RESULT

20ACB00978_FYP2.docx

ORIGINALITY REPORT

7%	5%	1%	3%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	eprints.utar.edu.my Internet Source	2%
2	Submitted to Universiti Tunku Abdul Rahman Student Paper	1%
3	Submitted to Nanyang Technological University Student Paper	<1%
4	dokumen.pub Internet Source	<1%
5	Jong-Moon Chung. "Chapter 3 XR HMDs and Detection Technology", Springer Science and Business Media LLC, 2023 Publication	<1%
6	flore.unifi.it Internet Source	<1%
7	www.tandfonline.com Internet Source	<1%
8	www.researchgate.net Internet Source	<1%

9	Ahsan Romadlon Junaidi, Yovie Alamsyah, Oktaviani Hidayah, Nur Wagis Mulyawati. "Development of Virtual Reality Content to Improve Social Skills in Children with Low Function Autism", 2020 6th International Conference on Education and Technology (ICET), 2020 Publication	<1 %
10	Submitted to CTI Education Group Student Paper	<1 %
11	Submitted to University of Southampton Student Paper	<1 %
12	dspace.daffodilvarsity.edu.bd:8080 Internet Source	<1 %
13	tandfonline.com Internet Source	<1 %
14	icli.um.ac.id Internet Source	<1 %
15	Submitted to South Gloucestershire and Stroud College Student Paper	<1 %
16	activation.unity3d.com Internet Source	<1 %
17	discussions.unity.com Internet Source	<1 %

18	Submitted to University of Wales, Bangor Student Paper	<1 %
19	Beibei Ye, Cikun Liu, Huan Li, Feifei Wang, Jian Chen, Yanbo Wang. "The design and application of xylose-lysine based time-temperature indicators for visually monitoring the shelf-life of chilled large yellow croaker", Journal of Food Engineering, 2023 Publication	<1 %
20	Submitted to University College London Student Paper	<1 %
21	animationxpress.com Internet Source	<1 %
22	docplayer.net Internet Source	<1 %
23	myfik.unisza.edu.my Internet Source	<1 %
24	staff.utar.edu.my Internet Source	<1 %
25	www.coursehero.com Internet Source	<1 %
26	Ton Duc Thang University Publication	<1 %
27	trepo.tuni.fi Internet Source	<1 %

28	Submitted to Pacific Lutheran College Student Paper	<1 %
29	documents.mx Internet Source	<1 %
30	slidetodoc.com Internet Source	<1 %
31	"Innovative Technologies and Learning", Springer Science and Business Media LLC, 2023 Publication	<1 %
32	Gupta, Shubh. "High-Integrity Urban Localization: Bringing Safety in Aviation to Autonomous Driving", Stanford University, 2023 Publication	<1 %
33	Michael Mortimer, Ben Horan, Matthew Joordens. "Kinect with ROS, interact with Oculus: Towards Dynamic User Interfaces for robotic teleoperation", 2016 11th System of Systems Engineering Conference (SoSE), 2016 Publication	<1 %
34	bonndoc.ulb.uni-bonn.de Internet Source	<1 %
35	ds.amu.edu.et Internet Source	<1 %
36	elibrary.tucl.edu.np Internet Source	<1 %
37	www.livescience.com Internet Source	<1 %

Exclude quotes	On	Exclude matches	< 8 words
Exclude bibliography	On		

PLAGIARISM CHECK RESULT

Universiti Tunku Abdul Rahman			
Form Title : Supervisor's Comments on Originality Report Generated by Turnitin for Submission of Final Year Project Report (for Undergraduate Programmes)			
Form Number: FM-IAD-005	Rev No.: 0	Effective Date: 01/10/2013	Page No.: 1of 1

FACULTY OF INFORMATION AND COMMUNICATION TECHNOLOGY

Full Name(s) of Candidate(s)	**TEE ZI JUN**
ID Number(s)	**20ACB00978**
Programme / Course	**Bachelor of Computer Science (Honours)**
Title of Final Year Project	**Customizable VR Design of Friends Interaction for Social Story**

Similarity	**Supervisor's Comments (Compulsory if parameters of originality exceeds the limits approved by UTAR)**
Overall similarity index: 7 **%** **Similarity by source** Internet Sources: 5 % Publications: 1 % Student Papers: 3 %	
Number of individual sources listed of more than 3% similarity: 0 %	

Parameters of originality required and limits approved by UTAR are as Follows:

- **(i) Overall similarity index is 20% and below, and**
- **(ii) Matching of individual sources listed must be less than 3% each, and**
- **(iii) Matching texts in continuous block must not exceed 8 words**

Note: Parameters (i) – (ii) shall exclude quotes, bibliography and text matches which are less than 8 words.

Note Supervisor/Candidate(s) is/are required to provide softcopy of full set of the originality report to Faculty/Institute

Based on the above results, I hereby declare that I am satisfied with the originality of the Final Year Project Report submitted by my student(s) as named above.

Signature of Supervisor	Signature of Co-Supervisor
Name: Ts Dr Goh Hock Guan	Name:
Date: 15/04/2024	Date:

UNIVERSITI TUNKU ABDUL RAHMAN

FACULTY OF INFORMATION & COMMUNICATION TECHNOLOGY (KAMPAR CAMPUS)

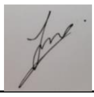
CHECKLIST FOR FYP2 THESIS SUBMISSION

Student Id	20ACB00978
Student Name	Tee Zi Jun
Supervisor Name	Ts Dr Goh Hock Guan

TICK (√)	**DOCUMENT ITEMS** Your report must include all the items below. Put a tick on the left column after you have checked your report with respect to the corresponding item.
√	Title Page
√	Signed Report Status Declaration Form
√	Signed FYP Thesis Submission Form
√	Signed form of the Declaration of Originality
√	Acknowledgement
√	Abstract
√	Table of Contents
√	List of Figures (if applicable)
√	List of Tables (if applicable)
	List of Symbols (if applicable)
√	List of Abbreviations (if applicable)
√	Chapters / Content
√	Bibliography (or References)
√	All references in bibliography are cited in the thesis, especially in the chapter of literature review
√	Appendices (if applicable)
√	Weekly Log
√	Poster
√	Signed Turnitin Report (Plagiarism Check Result - Form Number: FM-IAD-005)
√	I agree 5 marks will be deducted due to incorrect format, declare wrongly the ticked of these items, and/or any dispute happening for these items in this report.

*Include this form (checklist) in the thesis (Bind together as the last page)

I, the author, have checked and confirmed all the items listed in the table are included in my report.



____

(Signature of Student)

Date:15/04/2024