Technology Aided Learning Application for Wetlands Ecology using Constructivism Approach

NGWEE JIA LER

A project report submitted in partial fulfilment of the requirements for the award of Bachelor of Science (Honours) Software Engineering
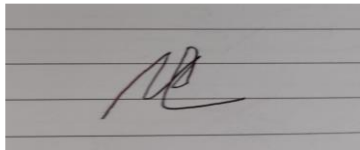
Lee Kong Chian Faculty of Engineering and Science
Universiti Tunku Abdul Rahman

Sep 2023

DECLARATION

I hereby declare that this project report is based on my original work except for citations and quotations which have been duly acknowledged. I also declare that it has not been previously and concurrently submitted for any other degree or award at UTAR or other institutions.

Signature   :

Name   :   NGWEE JIA LER

ID No.   :   2004069

Date   :   13/9/2023

APPROVAL FOR SUBMISSION

I certify that this project report entitled **"Technology Aided Learning Application for Wetlands Ecology using Constructivism Approach"** was prepared by **NGWEE JIA LER** has met the required standard for submission in partial fulfilment of the requirements for the award of Bachelor of Science (Honours) Software Engineering at Universiti Tunku Abdul Rahman.

Approved by,

Signature　　　　:

Supervisor　　　　:　　Ts Dr Lee Chen Kang

Date　　　　　　　:　　13/9/2023

Signature　　　　:

Co-Supervisor　：

Date　　　　　　　:

ACKNOWLEDGEMENTS

ABSTRACT

Wetland Ecology Education delves into the intricate relationships between wetland ecosystems. Due to a lack of awareness of the need for their conservation, wetlands are now fragmented and subject to economic development. To ensure the long-term sustainability of these ecosystems, several researchers and educational institutions have developed environmental education initiatives. The project addresses the main challenges of contemporary education. Emphasis is placed on learning as a constructive process where students actively engage with information and connect it to existing knowledge. By integrating technology and constructivist principles, the project aims to create immersive and authentic learning experiences within learning applications. The system includes five different modules that enable users to understand wetland ecology from all aspects. The most important of these is the simulation module, which contains 3D virtual environments and 3D models, allowing users to gain immersive experience and interaction. There is also a section responsible for collecting user feedback. This project is mainly developed using unity. Other development tools include Blender, phpMyAdmin, and Figma. After the project development was completed, SUS testing and PPSUQ testing were also conducted to ensure the quality of the application. In short, this project must be a new breakthrough for wetland education and can greatly improve the understanding of wetland ecology.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

1.1 General Introduction

Wetland ecological education is the study of the relationship between wetland ecosystems and the plants, animals and microorganisms that inhabit them. This area of research focuses on the processes that govern the functioning of these unique and complex ecosystems, including hydrology, nutrient cycling, carbon sequestration and biodiversity. Wetland ecology education also explores the impact of human activities on wetland systems, such as habitat loss, pollution, and climate change, and the mechanisms of wetland conservation and restoration.

Ecosystems are diverse, with essential functions and key services in chain interactions between organisms and their environment. Wetlands provide habitat and food sources for aquatic organisms, divert water flow during floods, purify water bodies, and affect food security (Ou, Chu and Tarng, 2021). Due to a lack of awareness of the need for their conservation, wetlands are now fragmented and threatened by the impact of economic development. To ensure the sustainability of the ecosystem, some researchers and educational institutions advocate environmental education, incorporating field trips and recreational activities into curriculum design to enhance environmental awareness and understanding of wetland ecology (Ou et al., 2021).

Educational institutions are re-evaluating the effectiveness of traditional teaching techniques and placing greater emphasis on effective strategies to enhance students' intellectual experience (Martín-Gutiérrez *et al.*, 2017). The virtual environment can beneficially change the education system. They open access to specialized information, previously inaccessible tools, and experiences that would otherwise be impossible (Christou, 2010; Greenwald et al., 2017). Ideas from a variety of fields can be taught and learned using technologies such as Augmented Reality (AR), Virtual Reality (VR), and Mixed Reality (MR) through interactive, differentiated, and

convenient experiences. The goal of these tools is to improve the outcomes that students achieve in the learning process (Fernandez, 2017).

In this report we will use digital media technology to enhance the learning experience in wetland ecology education and make it more engaging and interactive. Digital media technologies can provide students with new and innovative ways to explore and understand the complex relationships and processes that govern wetland ecosystems. Technology-assisted learning applications that simultaneously use constructivist approaches to teach wetland ecology may involve interactive simulations, virtual field trips, and educational games that allow students to actively engage and explore ecological concepts in wetland environments. This type of learning application can also include opportunities for students to construct their own knowledge by observing, analyzing data, and constructing models and explanations. The goal of the technology-assisted learning app is to support students to actively build their knowledge through meaningful and relevant experiences, thereby deepening their understanding of wetland ecology.

1.2 Importance of Study

Instruction today faces two critical challenges that have the potential to redefine the way we approach learning. These challenges arise from both changing perceptions of learning and new opportunities afforded by technology. The first challenge stems from constructivism, which presents the need to reconceptualize learning as a constructive process. This process involves interpreting information and actively relating it to existing knowledge, generating new representations, and purposeful elaboration (Maureen Tam, 2015). The second challenge is presented by the computer, which has opened up new learning opportunities that can help shift the focus of instruction from knowledge-as-possession to knowledge-as-construction, and from learning as outside-guided to learning as self-guided.

These shifts are highly congruent with the principles of constructivist teaching and learning. Constructivism and computing technology have significantly changed how we see learning's difficulties and have created new learning opportunities across all teaching and learning contexts, including face-to-face instruction, online learning, and

self-directed learning. Constructivism provides concepts and principles about education that have significant implications for the design of technologically enhanced learning environments (Maureen Tam, 2015).

One important implication is the need to embed learning into authentic and meaningful contexts (Maureen Tam, 2015). This approach requires students be involved fully in problem-tackling or decision-making cases. Case-based learning environments come in a variety of formats, such as cognitive flexibility, hypertexts, and anchored education. The fundamental idea behind these methods of instruction is to improve students' understanding and retention of information by exposing them to the same content in various situations and at various times for different purposes. (Maureen Tam, 2015).

1.3 Problem Statement

1) Wetlands ecology real field trips (RFTs) in addition to excessive labor costs, field trips are limited by the factors of time, space and the weather conditions.

RFTs by themselves cannot ensure that students will have a better learning experience because the majority of students lack background knowledge and introductory information about the actual plants or locations. Moreover, it may not be possible to have more than one RFT for each course due to constraints brought on by insufficient funding and resources for instruction as well as external pressure on faculties' time to conduct effective research (Krakowka, 2012). Fieldwork educators may be lacking in instructional techniques, which could reduce the effectiveness of RFTs (Hanson, 2011). Other difficulties include the lack of enough time to teach students everything they need to know, the difficulty of fieldwork educators' lessons, and the following partial transfer of practical knowledge (Seifan et al., 2019).

2) Lack of wetlands ecosystem learning application, conventional teaching materials are less interactive, so it is difficult to immerse students in the learning context.

The problem is the learning environment of the traditional classroom, which is teacher-centered, which makes the learning process tedious for less able students. Students' attention often declines, they are unable to retain information for long periods of time, and are often caught daydreaming, talking, and pestering other students (Li, 2016). Traditional classrooms usually ignore the differences in learning styles and personal abilities of each learner, which leads to poor learning effects.

Conventional teaching or traditional teaching refers to teaching methods in which teachers and students interact in a face-to-face manner in the classroom. These teachers initiate discussions in the classroom and focus on understanding what is in the textbook and notes. Many lecturers are still using traditional teaching and have noticed that in traditional teaching classes, when the lecturer explains and writes on the blackboard, students will copy the same content to their notes (Li, 2016). Traditional teaching also limits the space for more creative thinking and rarely takes individual differences into account. It is necessary to recognize these limitations in conventional teaching and take a step forward(Li, 2016).

3) Most of the existing applications have limitations (limited interactivity, lack of 3D virtual environment) to study and explore the wetlands ecosystem.

There are still many limitations in the existing learning applications, and there is still a big difference from the real wetland ecology. A common problem is that most of the current learning applications are based on 2D virtual learning environments, but the interactivity it provides is limited. For example, iBiome-Wetland (Spring Bay Studio, 2014) and Ecosystem Games (PBS kids, n.d.) learning applications both use 2d modeling as a virtual environment, and users can interact with Virtual environments to interact with, but they don't have the same level of physical presence or manipulation as the real world. A 2D virtual learning environment may not provide the same level of realism as a real environment, which can result in significantly less effective learning for learners. Some learning applications do not have any guided instruction, and the learner's first reaction is often not knowing how to start exploring and learning. These are all relatively common problems in existing learning application programs.

1.4 Aim and Objective

The aim of this project is to develop a learning application for target users to learn about the wetland ecosystem. The application allows target users to use online learning via the Internet, thus making learning more flexible and convenient. Target users can carry out immersive learning anytime and anywhere, so that target users can increase the basic knowledge and importance of wetland ecology in a simple way. The application includes 2-D and 3-D virtual environments and a constructivist approach, which is a major departure from traditional linear learning methods. The virtual environment can bring a more realistic experience to the target users, making the experience closer to the real world. Secondly, the constructivist approach can also enable more interaction and critical thinking between target users and applications.

1) To study the usage of digital media technologies (2-D and 3-D virtual environments) in aiding the education for wetlands ecology.

Student can be interactive with these digital media technologies, to learn how about the wetlands ecology proceed. Using this virtual environment to see the relationship between the animal, plants and environment and viewed from every angle. Because this is a 3D modeling so the effect will be more realistic, closer to the real world, and will also make a deeper impact on the user.

2) To design and develop the interactive learning application for wetlands ecology using constructivism approach.

Constructivism emphasizes the importance of active, experiential learning, which encourages learners to engage with concepts and ideas in a hands-on, interactive way. This approach helps learners to build a deeper understanding of the wetlands ecology material and make connections to their own experiences and prior knowledge. Constructivism promotes higher order thinking skills such as critical thinking, problem-solving, and creativity. By engaging in inquiry-based learning and exploration, learners develop the skills and habits of mind that are essential for success.

3) To evaluate the effectiveness of the learning application as compared to the traditional linear learning approach.

Learning application Compared with the traditional linear learning method, the learning method can have many forms, and there is no right or wrong. Each type of learner is different and has different expectations for their training experience. (LEPAYA PETER, 2021) Therefore, it is necessary to analyze the advantages and disadvantages of these two learning methods from various angles and evaluate which method can be loved by most people at present and the trend of future learning.

1.5 Target user

Choosing the public as the intended user for wetlands learning applications can have several advantages. Firstly, the public is made up of a diverse range of individuals with different degrees of interest in and knowledge about wetlands, and educational applications can aid in increasing awareness and fostering understanding among this group. Applications for learning about wetlands that are geared towards the public can reach a wide audience and advance a better knowledge of wetlands conservation and management. Learning applications can provide accurate and current information to help dispel misconceptions and promote a more informed of wetlands. Many people might not fully comprehend the ecological significance of wetlands or the threats they face. From public, wetlands learning applications can play an important role in promoting protect environmental and make informed decisions about wetlands conservation.

1.6 Proposed Solution

This project aims to solve the problems encountered in the current wetland ecological education. A learning application that will develop a web-based 3D virtual environment is proposed. The solution is essentially to integrate wetland ecology simulation and wetland ecology teaching materials into a web application, so that the general public can learn relevant information on the Internet. Let learners form more interactions in the learning process, and at the same time have more flexible and unrestricted learning.

There are 5 modules in this project, the most important function of which is the simulation module. This module will use a 3D virtual environment to build a wetland ecosystem, so that target users can interact with the model in a more direct way, and this experience will be closer to reality. The remaining modules also provide many learning materials to assist learning, so that target users can choose the learning methods they are interested in according to their preferences, and they can also have more different experiences.

The theoretical basis module provides the basic knowledge of wetland ecology, such as a brief introduction to the history and process of wetland ecology, the importance and the introduction of the illustrations of animals and plants. Although this approach is not much different from traditional teaching, it also proves its importance is indispensable. Secondly, the mini-game module and the evaluation module are considered to be the same type of learning method. Through some simple interactions, the target users will be more impressed with the learning knowledge and process. The game module is a relatively easy way for users to learn, and the attributes of entertainment and play are very important, so there are many ways to play. The evaluation mode is relatively serious. Users feel more like testing their own learning results when using this module, which is similar to school exams, so the selected topics need to be more rigorous. Finally, there is the feedback module. This module is to allow target users and the development team to have some communication methods, so that not only can we understand the various problems encountered by users during use, but also determine the shortcomings of the application and the direction of progress. In addition, social media accounts can also be created to facilitate target users to track and obtain relevant latest information and updates.

1.7 Project Scope

This project develops a functional software prototype (5 modules) that integrated with digital media technologies for 2-D and 3-D virtual environments for wetlands ecosystems.

The target users of this project are public. User can learn the wetlands ecologic knowledge, explore the relationship between wetlands animals, plants and environment using constructivism approach. The learning application include simulation module, theory base module, mini games module, Video introduction module, Story module. User can choose any five of this modules to learn if you are interest with that.
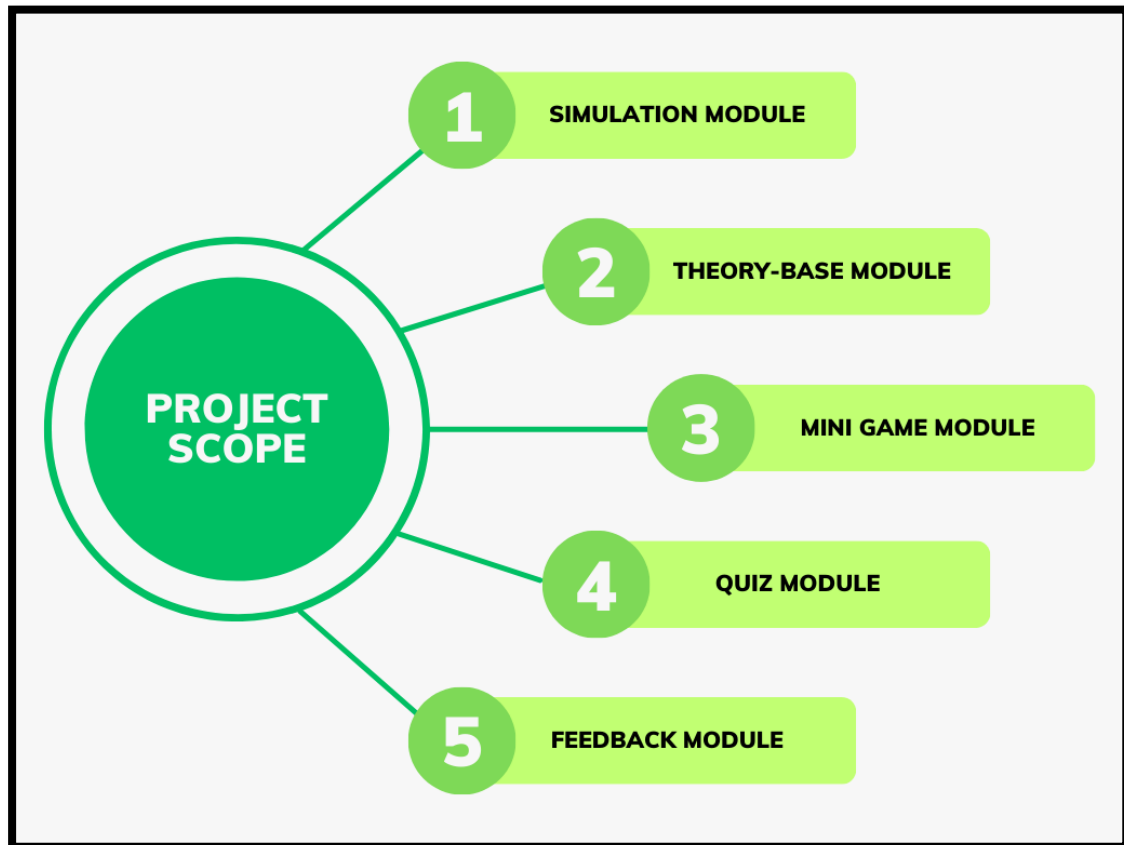
Figure 1.1 Project Scope 5 Module Diagram

1.7.1 Simulation module

Simulation module provides 3D virtual environments that simulate wetlands ecosystems, mimics real-world situation scenario allowing users to explore them. In addition to exploration, the module can be used to help users understand complex systems and concepts by allowing them to experiment with different variables and observe the outcomes.

In a wetlands ecosystem learning application, a simulation module could replicate various wetland settings, weather patterns, and human impact on the ecosystem. Users could change elements like the temperature, the weather, and the kinds of plants and animals present to observe how they affected the environment. The module may also provide examples of how human actions, such as pollution and habitat destruction, affect the ecosystem. With elements like animations, music effects, and real-time feedback, the simulation module may be made interactive and captivating.

1.7.2 Theory-base module

Theory-base module includes several sub-functions, including detailed information about wetlands ecosystems, an introductory video about wetlands, and infographic. The detailed information provides users with names, sizes, and other information about the animals and plants in the wetland. The introductory video is a video-based introduction to the wetland ecosystem, the video is not more than 10 min. A visual representation of data, information, or expertise is called an infographic. To convey complicated ideas in a straightforward and understandable manner, it frequently combines text with charts, graphs, icons, pictures, and other visual elements.

1.7.3 Mini games module

A mini-games module in a learning application provides users with a set of interactive games that are designed to help them reinforce and apply what they have learned. These games can be based on the content of the learning material or on specific skills that users need to develop. The games are designed to be engaging and fun, which helps to motivate users to continue learning. This module provides 2 to 3 games allows user to play around, for examples: Wetland Match, Wetland Food Chain and Wetland Puzzle.

1.7.4 Assessment Module

The Assessment module in a wetland ecosystem learning application is designed to evaluate users' knowledge and understanding of the material they have learned. The

module can include a variety of assessment methods, such as quizzes, tests, and assignments. In a wetlands ecosystem learning application, the assessment module can include questions that test users' knowledge of wetlands ecosystems, their components, and the relationships between them. The questions can be multiple-choice, true/false, or short answer, allows user think about the wetland's ecosystem process at the same time they will learn something on their creative thinking.

1.7.5 Feedback module

This module provides user feedback function, social media follow and FAQ (frequently questioned answers). Feedback function is used to describe the helpful information or suggestion and communicated to the development team. This function can let the development team easily understand how to adjust and improve the application. Besides that, this module provides the social media icon to let user easily follow our latest news and any updates. Last function is FAQ, user can review some of the frequently questions in here before send the feedback. This module objective is closer relationship with users.

CHAPTER 2

LITERATURE REVIEW

2.1 Wetland Ecosystem

2.1.1 What is Wetlands Ecosystem

A wetland is an area where its soil is covered by water all year or for varying periods of time. Wetlands are essential for human survival because they are the most productive ecosystems for providing aquatic and wildlife habitats, which are necessary for the survival of countless species of plants and animals. Additionally, they can enhance water quality, hold back floodwater, and keep surface water flowing during dry seasons. Therefore, a key component of environmental education has been the understanding of wetland ecology.

2.1.2 Why Wetlands ecology education is important

Wetlands ecology education is the study of the relationships between wetland ecosystems and the plants, animals, and microorganisms that inhabit them. This field of study focuses on the processes that govern the functioning of these unique and complex ecosystems, including hydrology, nutrient cycling, carbon sequestration, and biodiversity. Wetlands ecology education also explores the impacts of human activities on wetland systems, such as habitat loss, pollution, and climate change, and the mechanisms by which wetlands can be protected and restored. (Ou, Chu and Tarng, 2021)

2.1.3 How the value of wetland ecosystem

Wetlands were one of the first ecosystems to have their implicit values recognized in the early 1970s, likely because they had previously been thought of for such a long time, especially in the Western world, as systems that needed to be drained for human advancement (Mitsch, Bernal and Hernandez, 2015). In fact, people who studied other

ecosystems like grasslands, forests, and lakes experienced "ecosystem envy" as a result of the early identification and quantification of the many benefits of wetlands, particularly when those benefits were converted to monetary value. For example, the official mitigation of wetland losses in the 1980s in the USA through the creation and restoration of new wetlands has since been imitated for mitigation strategies for streams, floodplains, and even cropland. Both ecologists and economists should acknowledge and applaud the pioneering work done in demonstrating and quantifying the environmental services provided by wetlands(Mitsch, Bernal and Hernandez, 2015). Now more than ever, it's critical to remember the benefits wetlands offer.

Wetland ecosystems in India sustain different and unique habitats and are located across various geographical and climatic regimes. They are extremely productive systems in their natural forms and are regarded as an essential component of the hydrological cycle. Wetlands offer a vast range of ecosystem goods and services in addition to supporting a diverse range of biological species(Bassi et al., 2014).

2.2 Virtual Learning Environment

2.2.1 Definition of VLEs

Information and communication technologies (ICT) have become an essential part of modern-day education, revolutionizing the way students learn and interact with educational content. As a result, it has become increasingly important for higher educational institutions (HEIs) to collaborate with ICT to create innovative and effective learning experiences for their students. One way in which this collaboration can take place is through the use of virtual learning environments (VLEs). VLEs are computer-based platforms that allow students and educators to interact with educational content in a variety of ways, including online discussion forums, virtual classrooms, and multimedia content (Wan Abdul Rahim Wan Mohd Isa, 2019). By using VLEs, HEIs can create a more immersive and engaging learning experience for their students, as well as provide a more flexible and convenient way of accessing educational content.

2.2.2 Why VLEs are gradually replaces traditional learning

In her study of students' views on the changes higher education has undergone, O'Donnell (2012) focused on the use of technology in VLE. According to the study and the student responses, it was evident that students thought that the use of technology in higher education positively changed learning. Despite this, using technology for learning will never take the position of a human teacher. In summary, how technology is used pedagogically by a teacher will determine the benefits that can be attained through its use. The students in this study already possessed information technology and communication skills, meaning they were able to use social media successfully, according to a review of articles and studies on virtual learning environments. Traditional learning is replaced and encouraged by VLE. Education nowadays relies on students being able to use online learning applications effectively. (Phungsuk, Viriyavejakul and Ratanaolarn, 2017)

2.2.3 How the 3D VLE show their advantages

3D virtual learning environment is a new form of network learning environment. The 3D virtual learning environment has intrinsic advantages in the virtual learning environment and can better support network inquiry be learning and cooperative learning because it has the benefits of high visualisation, strong interest, and diversified content presentation. The bottleneck limiting the growth and use of computer technology will no longer be computer hardware or network bandwidth. A 3D virtual learning environment will eventually turn into the best setting for network learning in the future and a potent addition to the traditional learning environment. (Wang, 2020)

2.3 Constructivism Approach

2.3.1 What is Constructivism Approach

Constructivism is essentially a theory about how people learn that is founded on observation and research. According to this, individuals build their own knowledge and understanding of the world by engaging in experiences and thinking back on them

(Bereiter, 1994). The constructivist theory of learning can suggest a variety of instructional strategies in the classroom. In its broadest meaning, it typically refers to motivating students to use active learning strategies (experiments, real-world problem solving) to expand their knowledge, and then to reflect on and communicate how their understanding is evolving. The instructor ensures that he or she is aware of the students' preconceived notions and directs the activity to address and then expand upon them (Oliver, 2000).

Constructivism is a philosophy that fosters pupils' conceptual and logical development Driscoll (2000). The central idea of the constructivism learning theory is the importance of experiences and connections to the surrounding environment for pupil education. According to the constructivism learning theory, individuals create knowledge and create meaning as a result of their experiences.(Olusegun, no date) Kim proposed that people build new knowledge from their encounters through the processes of accommodation and assimilation. (Kim, 2005)

2.3.2 The process of Constructivism Approach

The accommodation process, which is the initial step in the production of knowledge, reframes the world and new experiences to match the person's existing mental capability. When something does not function within that environment, it must be accommodated and the expectations and results must be reframed. As a result, the learner makes room in the mental faculties where the old experiences are already seated for the new experiences that they have acquired (Olusegun, no date; Kim, 2005). Contrarily, in the second and final stage of the knowledge-building process, called assimilation, the person integrates the new experience into an existing framework of prior experiences without altering that framework. Thus, this prompts the person to adopt fresh perspectives, reconsider earlier misconceptions, and assess what matters most, ultimately changing their perceptions (Kim, 2005; Olusegun, n.d.).

2.3.3 How the Constructivism Approach benefits learning

Theoretical foundations for instructional design include constructivism, behaviourism, and many theoretical frameworks. Constructivism is the preferred instructional design paradigm since sustainability is a complicated, multifaceted subject that has a steep learning curve. The conceptual structure of the study is based on constructivism, according to which learning occurs through interaction with the environment rather than through passive acquisition. The emphasis is consequently on actively creating meaning from one's environment as opposed to passively receiving it from them. De facto, teaching sustainability through constructivism necessitates a hands-on strategy where students gain knowledge through doing and participating. They obtain a deeper awareness of the intricacy of the topic by actively participating in sustainability challenges, as sustainable development is inherently complex and difficult (Hedden et al., 2017).

If constructivist theory is the best approach to define learning, it follows that in order to encourage student learning, learning environments that expose the learner directly to the subject matter must be created. Because the learner can only understand things by experiencing them immediately. This supports the idea that constructivist learning has to occur in a conducive learning environment. Any constructivist learning environment must allow for active learning because it is one of the fundamental tenets of all constructivist learning that it must be an active process (Olusegun, n.d.).

2.4 Multimedia

2.4.1 The definition of multimedia technology

Multimedia is a combination of two words, multi and media, which refers to the use of various types of media (hardware and software) for the communication of information. It is the interactive and entertaining depiction of information using a mix of text, audio, video, graphics, and animation. A computerized method of presenting information that incorporates textual data, audio, image, video, graphics, and animations is referred to as multimedia (*Multimedia Introduction,* no date).

Multimedia includes a number of widely used components. First of all, text is a component of all multimedia presentations and can be presented in a variety of fonts

and sizes to match the professional presentation of the multimedia programme (*Multimedia Introduction*, no date). Secondly, graphics are a significant aspect of multimedia applications as they make the presentation more appealing. Graphics are used to explain a concept, provide background information, and other things instead of reading a lot of words. The two forms of graphics utilized in multimedia are bitmap and vector (*Multimedia Introduction*, no date). While vector graphics are generated on the computer and only use a little amount of memory, bitmap images are actual images that have been recorded by tools like digital cameras or scanners (*Multimedia Introduction*, no date).

Another critical component of multimedia is audio, which includes speech, music, and sound effects. The two types of audios used in multimedia applications are analog and digital. While the computer saves sound in digital form, analog audio or sound refers to the original sound signal. A crucial component of multimedia is digital video, which is the term for the moving image and audio that is identical to the image on television (*Multimedia Introduction*, no date). It is helpful for displaying real-life items in multimedia applications since it can convey a lot of information in a short amount of time. Lastly, animation is the process of making a static image appear as if it is moving. It is a continuous series of still photos that are shown in a sequence and can be utilised to attract attention and improve a presentation (*Multimedia Introduction*, no date).

2.4.2 The importance of multimedia in education

The first principle of interactive multimedia learning involves integrating various media components into the learning experience. Interactive multimedia courseware is a computer-based learning tool that combines text, images, photos, audio, video, and animation into a single product (Sebelas and Rejekiningsih, 2020). The choice of which elements to use depends on the specific material being developed. These multimedia elements include text for explanations and instructions, graphics for illustrations and simulations, video for real-life object simulations, animation for moving images, and audio for background music, narration, sound effects, or conversation (Sebelas and Rejekiningsih, 2020).

The second principle of interactive multimedia learning involves navigation and interactivity. Interactive multimedia learning contains links that enable users to interact with the media. It provides navigation tools to facilitate user-directed learning and offers feedback. The learning experience is controlled by a controller that gives students the freedom to determine the course of multimedia learning (Sebelas and Rejekiningsih, 2020).

2.4.3 Development of educational multimedia modules

Multimedia software should be interactive, motivating, and relevant, with lots of action and novelty (Stemler, 1997). These general suggestions, nevertheless, fall short for educators looking for more focused advice. Gagne's nine events of learning, which were originally created for teaching ideas to people, fortunately offer a framework for creating effective educational multimedia modules. Gagne's nine events have been suggested by multimedia researchers as a technique to make sure that programmes are effective (Stemler, 1997). Gaining attention, explaining the lesson's goals and igniting motivation are a few of these events. Others include presenting the stimulus material, offering learning direction, eliciting performance, offering feedback, assessing performance, and improving retention and learning transfer. Numerous multimedia aspects enhance the internal learning process and correspond to Gagne's events.
.Thus, following Gagne's nine events can help educators create effective multimedia programs (Sebelas and Rejekiningsih, 2020).

2.4.4 Summary of multimedia

The use of multimedia in presenting information offers a plethora of exciting possibilities by combining hypermedia techniques with instructional materials. The creation of good presentations involves basing them on cognitive objectives that are geared towards learning topics at different levels of comprehension (Stemler, 1997). It may be difficult to determine the extent to which a single multimedia component, such as graphics, menus, audio, or video, contributes to learning outcomes. There are three fundamental aspects that designers can try to enhance: (a) capturing the learner's attention, (b) assisting learners in finding and organizing pertinent information, and (c) integrating this information into their existing knowledge structures (Stemler, 1997). Rather than focusing on individual text elements, multimedia developers should

instead concentrate on arranging text, graphical, and audio-visual components in a manner that creates organized, structured, and visually appealing screens.

2.5 Similar application Research

2.5.1 Application 1: Natural Selection (Nobel Laureate Carl Wieman, 2002)

link: https://phet.colorado.edu/en/simulations/natural-selection

The Natural Selection simulation by the PhET Interactive Simulations project is an educational tool designed to help students understand the principles of natural selection, a fundamental concept in evolutionary biology. In this simulation, students can build a variety of settings and study how different populations of organisms evolve over time because of factors such as genetic variation, natural selection, and environmental stresses. Students can modify parameters like as mutation rate, selection pressure, and population size to understand how these factors effect population evolution.

Strengths: The simulation allows students to actively engage in the learning process by manipulating variables and observing the effects of their actions. The simulation allows students to adjust various parameters, such as mutation rate, selection pressure, and population size, to see how these factors affect the evolution of populations show in Figure 2.1. This provides a more nuanced and dynamic understanding of evolutionary processes. Both a data view and a simulation or animation view are available in this study show in Figure 2.1. Users can more easily and plainly see the variation in the data-to-data views, which is helpful when examining the relationship.

Weakness: The simulation requires a computer or mobile device with internet access, which may not be available to all students. Therefore, the design is too basic, a little out of date, and not as engaging as models of real-world situations. Because this lab lacks the instructions to guide students on how to use it, it is very challenging for students to be self-guided, necessitating teacher input and guidance.
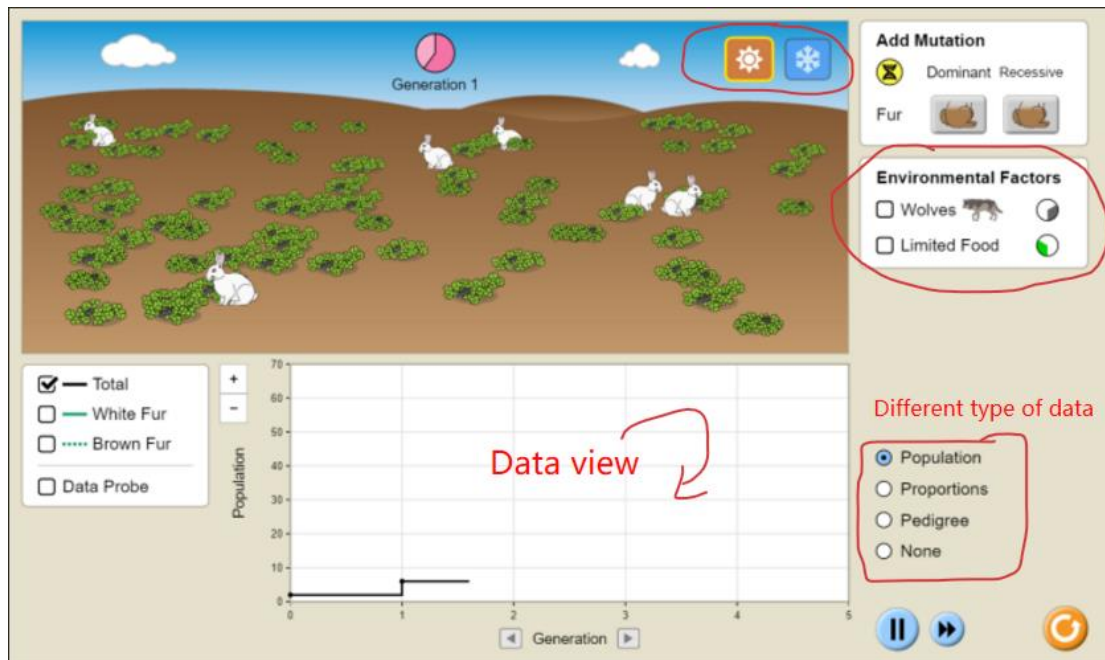
Figure 2.1 Natural selection web app

2.5.2 Application 2: Equilinox (ThinMatrix, 2018)

Equilinox is a simulation game developed by ThinMatrix, a solo game developer based in the United Kingdom. This is a relaxing nature game which allows you to create and nurture your very own diverse ecosystem of plants and animals (ThinMatrix, 2018). Players can manipulate the environment by adjusting factors such as temperature, rainfall, and soil fertility to create different habitats and attract different species of plants and animals. As the ecosystem grows and evolves, players can observe the interactions between species, manage food chains, and maintain the delicate balance of the ecosystem.

Strengths: Equilinox has been praised for its beautiful graphics show in Figure 2.2, and educational value in teaching players about the complexities of ecosystem management. You can create and maintain your own particular ecosystem of flora and animals in the serene nature game Equilinox show in Figure 2.3. Every animal and plant has a unique life cycle, distinct behaviours, and requirements show in Figure 2.4, so you must ensure that each species is happy with its surroundings, has access to enough resources, and that the ecosystems are balanced. This program's creatures, vegetation, and habitats were developed using three-dimensional modelling. One of

the benefits of 3D environments for learning is their ability to help us comprehend the complicated systems we encounter in the real world, such as the ecological system. (ThinMatrix, 2018)

Weakness: Although Equilinox uses 3D modelling, the models are only low-quality 3D models, so they lack realism. Since this is a foster game, we need some time to examine the ecology and see the outcome.
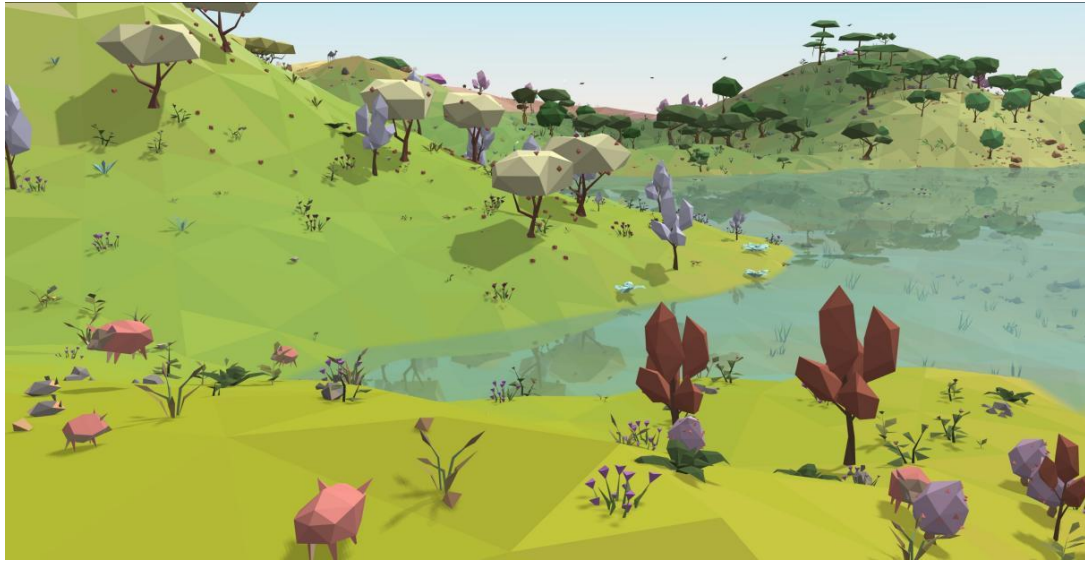
Figure 2.2 Equilinox 3D virtual environment

Figure 2.3 Equilinox building system

Figure 2.4 Equilinox model status

2.5.3 Application 3: Ecosystem Simulation (Claireyuke, 2021)

Link: https://claireyuke.itch.io/ecosystem-simulation

Ecosystem simulation and procedural village generation are the topics of this undertaking. The project mimics a wildlife forest where users can grow plants to feed animals while also watching them go about their daily lives and engage in reproduction and hunting. Players can select to experience various atmospheric

conditions in the meantime. In addition to the forest, a procedural town generation process is present. The hamlet is home to a variety of farm animals. Players can alter the village's building specifications and get a sense of what it's like to live with animals.

Strengths: Ecosystem Simulation is a relatively simple game, with not many functions but it provides a reality model and environment for players to learn about ecology. As the ecosystem grows, players must manage factors such as food availability, water supply, and temperature to ensure that all species can thrive show in Figure 2.6. Players can also observe and interact with the species in their ecosystem, watching as they hunt, mate, and compete for resources show in Figure 2.7.

Weakness: Because this is an undergraduate project compared to other ecosystem simulation games, Ecosystem Simulation is relatively simple and has fewer features. This may limit its usefulness for advanced or more experienced players who are looking for a more complex simulation. Besides that, the game has a limited number of species and habitats to choose from, which may make it less engaging for players who are looking for more customization options.

Figure 2.5 Ecosystem Simulation Home

Figure 2.6 Ecosystem Simulation eat and drink

Figure 2.7 Ecosystem Simulation 3D modeling simulation

2.5.4 Application 4: iBiome-Wetland (Springbay Studio Ltd, 2014)

iBiome-Wetland is an enjoyable way for young people to educate themselves about various wetland ecosystems. As they add a variety of approved species to their biodomes, they'll be able to experiment and explore at their own speed. Children gain knowledge of producers, consumers, and predator/prey relationships by connecting various species and building a web. They will, however, need to work quickly because the exercises are timed, and students "lose" if they fail to draw the webs in a timely manner. For students who need extra time to complete the chores, it would be convenient to have the timer turned off. The game's journal component would benefit greatly from a glossary of scientific terms (Debbie G., 2014).

Strengths: The app has impressive graphics and animations that make wetland learning enjoyable and fun. The app makes a short video introduction to help users better understand the wetland ecology before starting the study show in Figure 2.8. A diary allows learners to keep track of species discovered and encourages exploration. Mini-games provide an interactive game-based learning experience to teach students about wetlands show in Figure 2.10 (Spring Bay Studio, no date).

Weakness: The app doesn't have a free version, is a paid app, which may limit its accessibility for some users and educators. The app is only available on iOS devices, which may limit its accessibility for users who prefer other platforms. The process of adding species to biomes may be confusing and needs more detailed instructions and specific outcomes (Debbie G., 2014).

Figure 2.8 iBiome-Wetland Introduction

Figure 2.9 iBiome-Wetland System

Figure 2.10 iBiome-Wetland Mini game

2.5.5 Application 5: Ecosystem Games (PBS Kids, 2015)

Link: https://pbskids.org/plumlanding/games/ecosystem/jungle_jeopardy.html
This game allows players to see how the different species of plants and animals in a rainforest depend on one another, and to experiment with how changing the amount of one resource affects the whole ecosystem. This game developed by pbs teams on their learning website.

Strengths: The game is free and can be played by anyone with an internet connection. The game is interactive and engaging, with colorful graphics and sounds that help keep kids interested. A video introduction and some gameplay instructions are provided before the game starts show in Figure 2.11. In addition, this game also has some sound feedback for user operations, so that users will not be too bored during the game.

Weakness: The game is designed for younger children and may not be challenging enough for older children or those with more experience in learning about ecosystems. The game does not provide detailed feedback on incorrect answers, which may limit its effectiveness in helping children learn.

Figure 2.11 Ecosystem Games Introduction and Instruction

Figure 2.12 Ecosystem Games Building System

2.5.6 Comparison Table for five similar applications

Table 2.1 Compare Similar Applications

	Application	2D/3D Modeling	Creativity /Interaction	Instruction / Intro	Mini Games	Learning Time
Natural Selection (Nobel Laureate Carl Wieman, 2002)	Web	2D	Weak	No	No	Short
Equilinox (ThinMatrix, 2018)	Steam	3D	Strong	Yes	Yes	Very Long
Ecosystem Simulation (Claireyuke, 2021)	Web	3D	Weak	No	No	Short

iBiome-Wetland (Springbay Studio Ltd, 2014)	Mobile	2D	Weak	Yes	Yes	Short
Ecosystem Games (PBS Kids, 2015)	Web	2D	Normal	Yes	Yes	Short

Table 2.2 Compare similar applications with multimedia elements

	Text	Graphic	Image	Audio	Video	Animation
Natural Selection (Nobel Laureate Carl Wieman, 2002)	Yes	Yes	Yes	No	No	Yes
Equilinox (ThinMatrix, 2018)	Yes	Yes	Yes	Yes	No	Yes
Ecosystem Simulation (Claireyuke, 2021)	Yes	Yes	Yes	No	No	Yes
iBiome-Wetland (Springbay Studio Ltd, 2014)	Yes	Yes	Yes	Yes	Yes	Yes
Ecosystem Games (PBS Kids, 2015)	Yes	Yes	Yes	Yes	Yes	Yes

2.5.7 Conclusion

In conclusion, the five learning apps aimed at ecological education each have their own unique methods and designs, with varying strengths and weaknesses. The most important factor in these apps is user-friendliness, which includes ease of use, navigation, and understanding. Cumbersome apps can be confusing and frustrating, while user-friendly apps increase engagement, retention, motivation, and time efficiency, all of which are crucial for effective learning. UI design is also crucial, as it enhances the overall user experience, making an application more attractive, intuitive, and user-friendly. 3D modeling provides a more realistic and engaging virtual environment, but it can take longer to explore. Background music and audio feedback are also important, as they can increase interactivity and make users more relaxed. However, the lack of theory-based knowledge and guided tutorials in some apps is a downside. Additionally, learning materials must be appropriately challenging and not too short or too long to maintain user engagement. Overall, the success of a learning app depends on a combination of these factors, and it's important to consider them carefully when designing such an application.

2.6 User Requirement Gathering and Analysis

2.6.1 Survey Questionnaire

In this report, the author conducted a user learning application survey using a questionnaire to collect data from the target users and to identify their preferences. Using a standardised set of questions, a questionnaire is a research tool that gathers data from a sample group. There are numerous ways to conduct questionnaires, including through interviews, internet surveys, and paper-based forms. Open-ended or closed-ended questions can be used to gather data on a variety of issues, such as demographics, attitudes, opinions, behaviors, and preferences.

Numerous factors make questionnaires significant. They can be used to gather a lot of information from a large sample population in order to create a representative sample of a target user. Additionally, by guaranteeing that all respondents are asked the same questions in the same way, standardising questionnaires helps to minimise the risk of

researcher bias or inaccuracies. Additionally, questionnaires can be used to collect both qualitative and quantitative data, providing researchers with a diverse set of information to analyse.

To provide a quick and effective data collection process, we used Google Forms as the survey platform. Google Forms is a powerful and simple tool for creating customised questionnaires that can be readily disseminated to a large number of participants. First of all, in order to improve data analysis, most of our questions are closed-ended. Section A of this Google form contains inquiries for demographic data, while Section B concerns learning application modules. Additionally, Google Forms can automatically assess Additionally, Google Forms has the ability to automatically analyze the information gathered from the survey and create graphs, which aids in visualizing the outcomes and better illuminating the preferences of the participants. Appendix A will show the Google form questionnaire screenshot.

2.6.2 Data Collection and Analysis
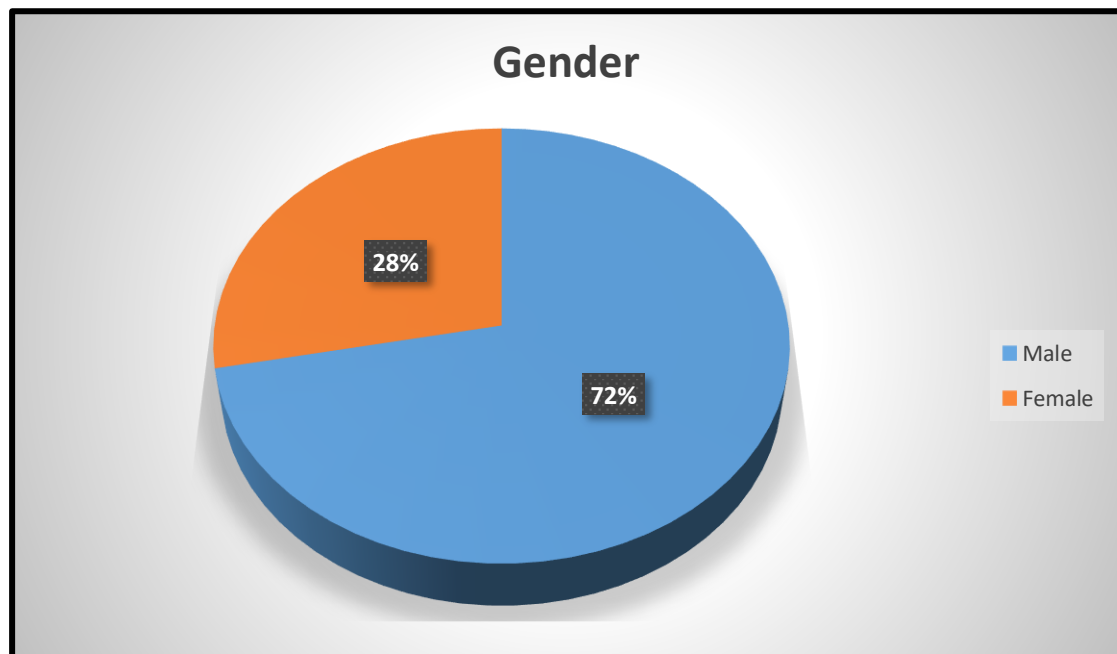
Figure 2.13 Pie chart of gender

The pie chart above shows the percentage of gender from the 32 responses. Majority of them are male the percentage is as high as 72%, female only have 28% in this questionnaire.
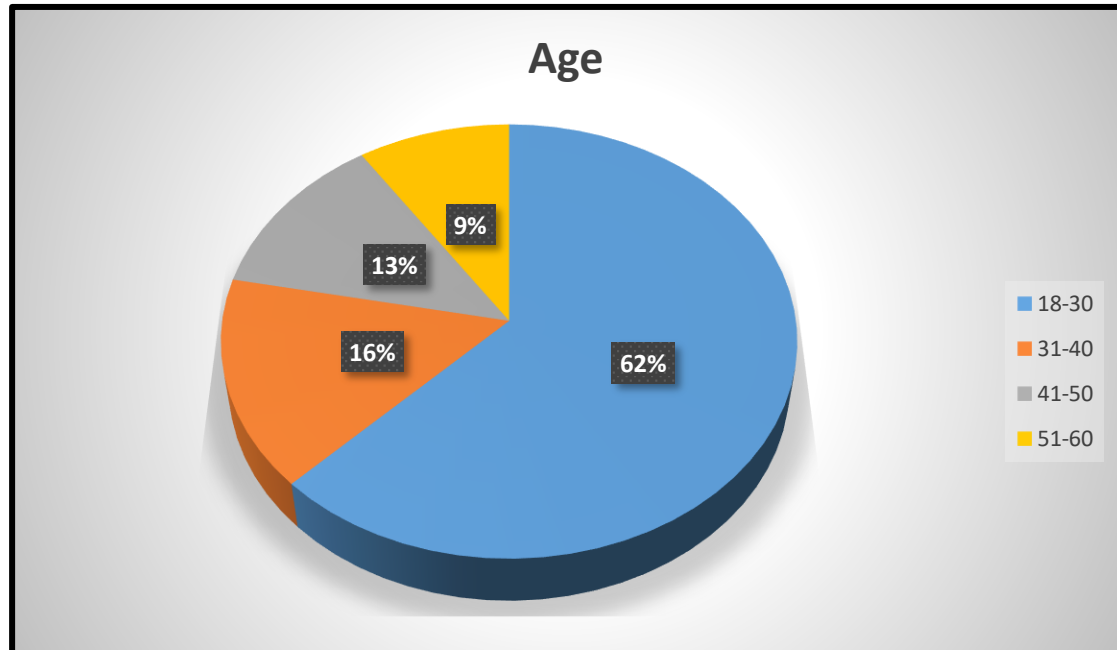
Figure 2.14 Pie chart of age

The pie chart above shows the percentage of age from the 32 responses. Majority of them are 18-30 years old the percentage is as high as 63%. There are 16% of respondents are 31-40 years old, 12% of respondents are 41-50 years old and only have 9% of respondents are 51-60 years old.

Q1. Using learning applications can enhance your understanding and knowledge.
32 responses

Figure 2.15 Column chart of Q1

The column chart above shows the number and percentage of Q1 from the 32 responses. Majority of them are agree with that using learning applications can enhance your understanding and knowledge. 15 respondents agree and 9 respondents strongly agree with that total is 75% and no respondents disagree learning application can help user improve knowledge.
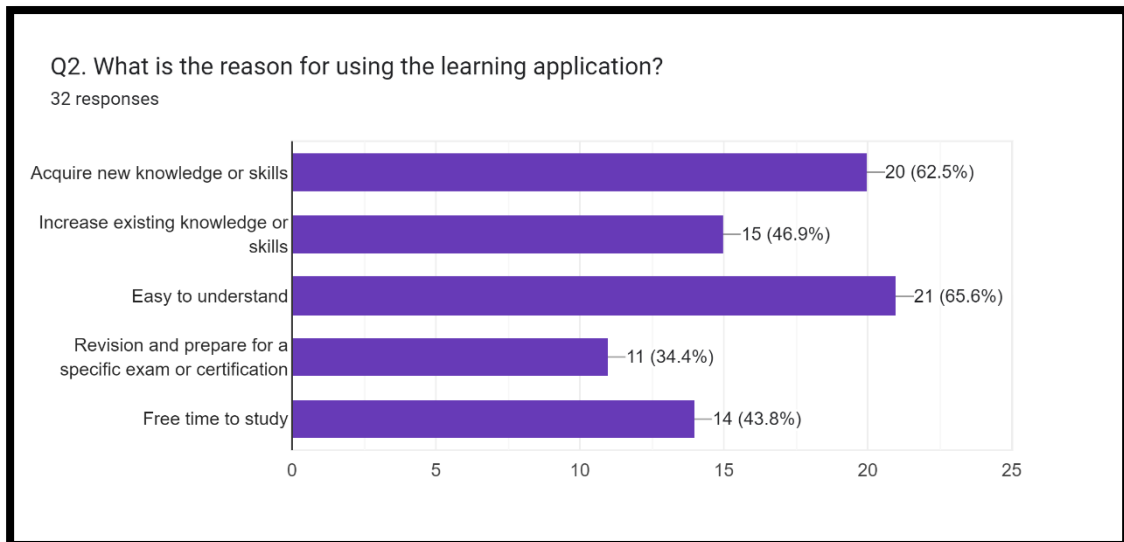
Figure 2.16 Bar chart of Q2

The bar chart above shown that most of the response of the reason for using the learning application. The Large number of respondents for using the learning application are easy to understand and acquire new knowledge and skills. Learning app content compared to online source will be quick easy to understand and more suitable with beginners acquire new knowledge. Other three reason less than 50%, there are increase existing knowledge or skills (46.9%), free time to study and revision (43.8%) and prepare for a specific exam or certification (34.4%).

Q3. What type of digital simulation you are prefer to use?

32 responses

3D virtual environment interactive simulation	23 (71.9%)
2D virtual environment interactive simulation	10 (31.3%)
Graphic simulation	10 (31.3%)
Video simulation	18 (56.3%)
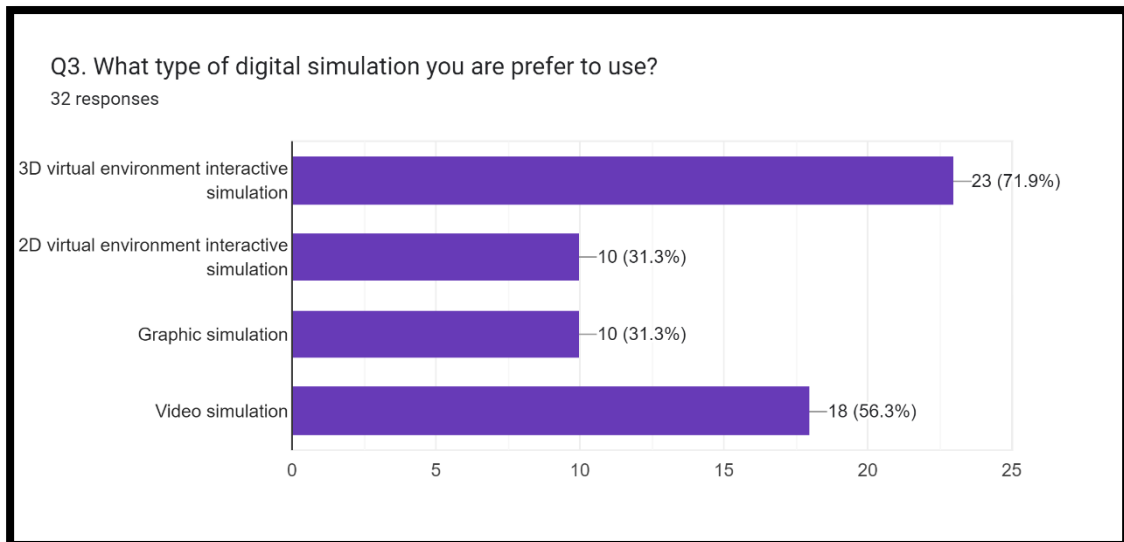
Figure 2.17 Bar chart of Q3

The bar chart above shown that most of the response of Q3 what type of digital simulation you are preferring to use. The most of response for the type prefer to use in learning application are 3D virtual environment interactive simulation. 3Dsimulation is a new technical and it will more realistic and interactive so user will be interested. The Large number of respondents prefer to use video simulation because video simulation is quick relax user not need put more attention than they can learn something. Learning app content compared to online source will be quick easy to understand and more suitable with beginners acquire new knowledge. 2D virtual environment interactive simulation (31.3%) and Graphic simulation (31.3%) only have 10 respondents to choose maybe is quick common.
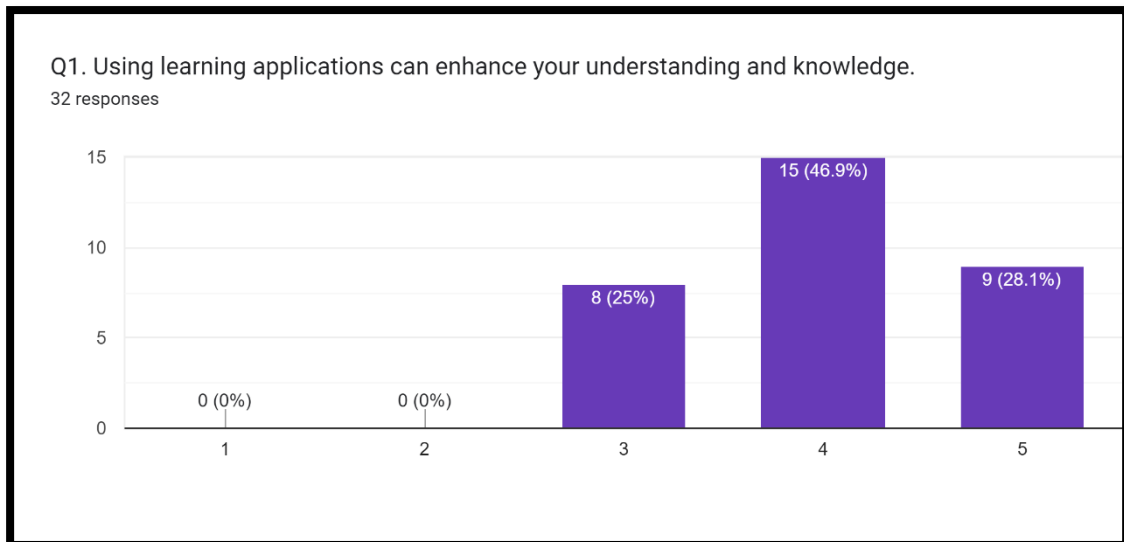
Figure 2.18 Column chart of Q4

The column chart above shows the number and percentage of Q4 from the 32 responses. Majority of them are agree with that the utilization of a 3D virtual environment can provide a highly interactive, realistic, and effective approach to learn about wetland ecosystems. 12 respondents agree and 9 respondents strongly agree with that total is 65.6%, 10 respondents stand neutral, and 1 respondent disagree 3D virtual environment can get a good effective.

Q5. Which of the following features would you interested in wetlands simulation module? Choose at least two.
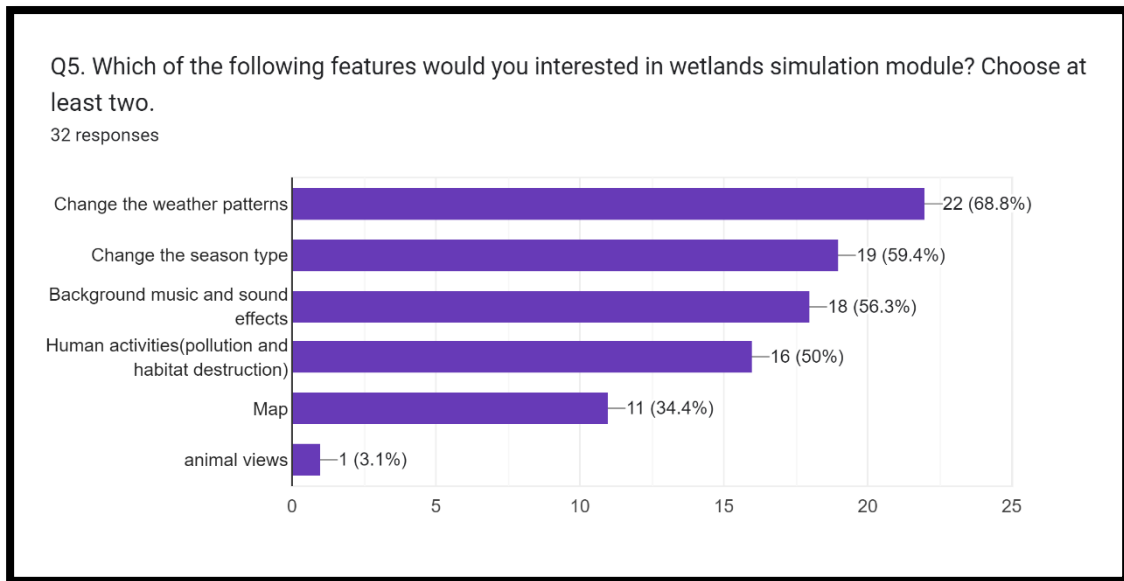
32 responses

Figure 2.19 Bar chart of Q5

The bar chart above shown that most of the response of Q5 which of the following features would you interested in wetlands simulation module. The most of response for the interest features to use in learning application are change the weather patterns. The Large number of respondents interest feature change the season type, background music and sound effects and human activity. Map features is not needing focus only have 11 respondents choose that and what of the respondents recommend animal views, this is a good idea I lost from the questionnaire.
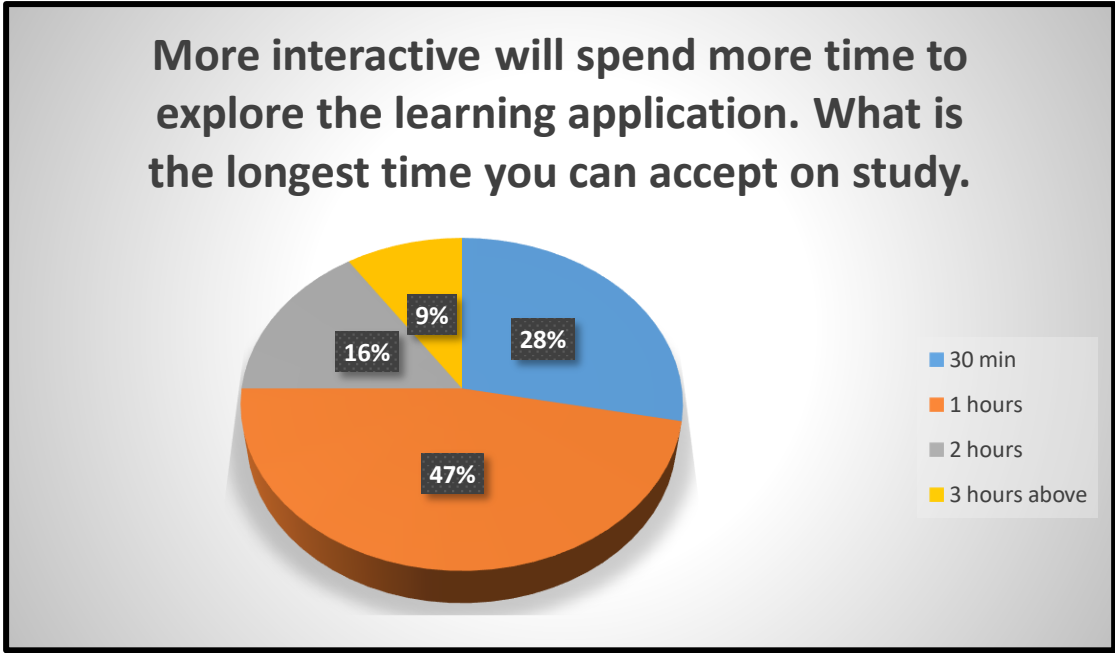
Figure 2.20 Pie chart of Q6

The pie chart above shows the percentage of age from the 32 responses. Majority of them are preferring to spend 1hours to using the learning application. 28% of the respondents just prefer to use 30 minutes to explore. There are 16% of respondents prefer to spend 2 hours to learning, but only 9% of respondents willing to pay 3 hours and above to explore the learning application.

Figure 2.21 Column chart of Q7

The column chart above shows the number and percentage of Q7 from the 32 responses. Majority of them are agree that mini games allow users to learn and play

together, helping to motivate users to keep learning. 13 respondents agree and 8 respondents strongly agree with that total is 65.6%, 11 respondents stand neutral, and no respondents disagree mini games can help user to keep learning.

Q8. Which of the following type of game would you prefer in mini games module? Choose at least two.

32 responses

Game	Responses
Puzzle Game	20 (62.5%)
Spelling Game	13 (40.6%)
Match Game	24 (75%)
Food Chain Game	14 (43.8%)

Figure 2.22 Bar chart of Q8

The bar chart above shown that most of the response of which of Q8 the following type of game would you prefer in mini games module. The most of response for the interest mini game in learning applications are match game. The Large number of respondents interest mini game is puzzle game. Food pyramid and spelling mini game only have 14 and 13 respondents are interest. Food pyramid user not interested may be is not understanding the meaning this is a big issue will affect the data. Spelling game we can learn at the match game, it is quite about the basic knowledge and theory, user won't like it at the mini game module.
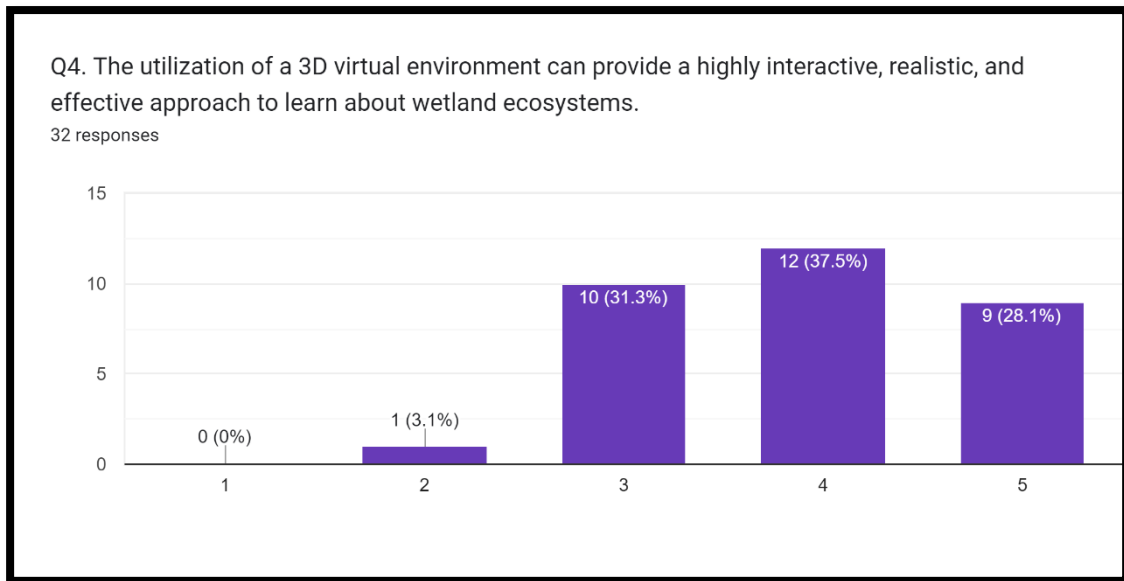
Figure 2.23 Column chart of Q9

The column chart above shows the number and percentage of Q9 from the 32 responses. Majority of them are agree that Learning assessment attract user in the wetland ecosystem learning application. 16 respondents agree and 5 respondents strongly agree with that total is 65.6%, 11 respondents stand neutral, and no respondents disagree assessment can attract user to keep learning.

Figure 2.24 Bar chart of Q10

The bar chart above shown that most of the response of Q10 Which of the following type of questions would you prefer in assessment module. The most of response for

the prefer assessment type in learning applications are Multiple choice questions and true false questions. These two types of assessment compared to other questions will simpler and easier to answer because the choice already provide to user. Short answer and fill in the blank questions only have 14 and 7 responses may this two type questions will spend a lot of time and don't have the correct answer.

Q11. In assessment module, which of the following type of feedback would you like to received after answering the questions.

32 responses

Feedback Type	Responses
Give the correct answer	15 (46.9%)
Explanation of the correct answer	25 (78.1%)
Marking(Scores)	16 (50%)

Figure 2.25 Bar chart of Q11

The bar chart above shown that most of the response of Q11 which of the following type of feedback would you like to receive after answering the question in assessment module. The most of response for the prefer assessment feedback in learning applications are explanation of the correct answer. The Large number of respondents prefer assessment feedback are marking and give the correct answer. In these questions most of the user prefer the assessment feedback will give the correct answer and the explanation and also the marking then the user can clearly check their result and why this is the correct answer.
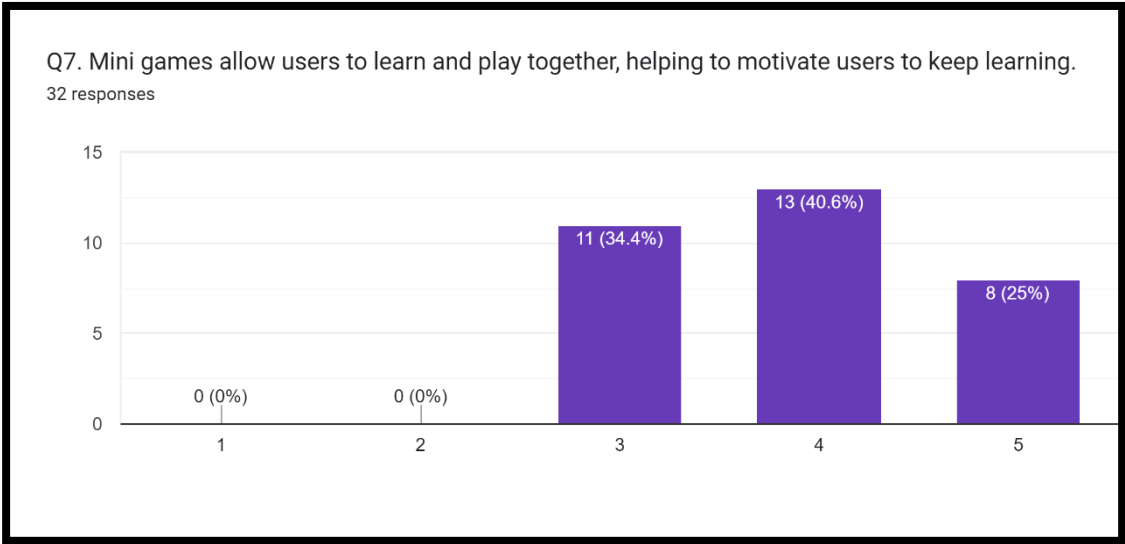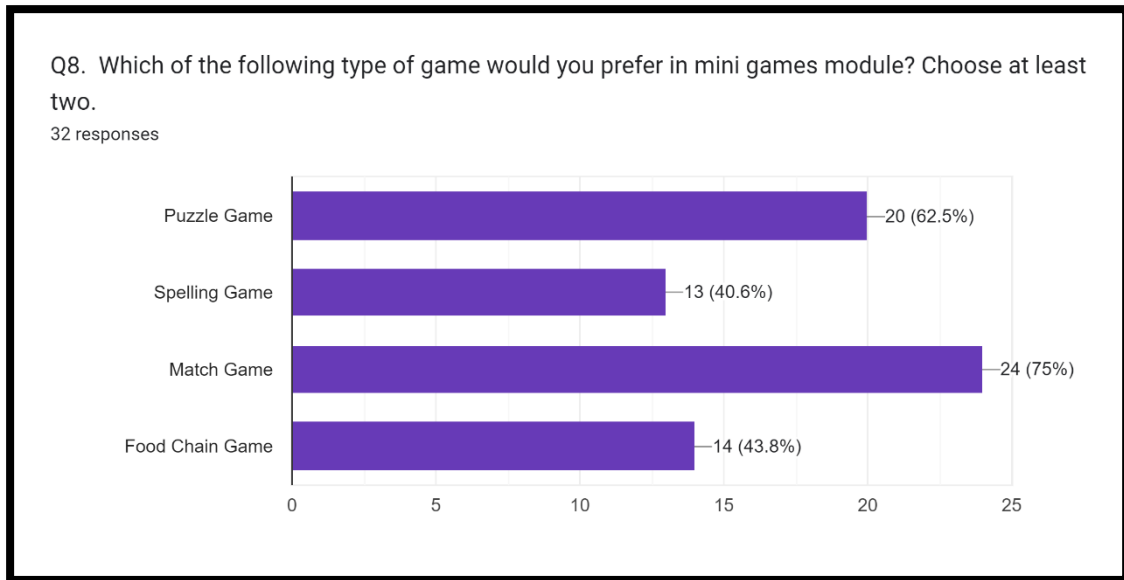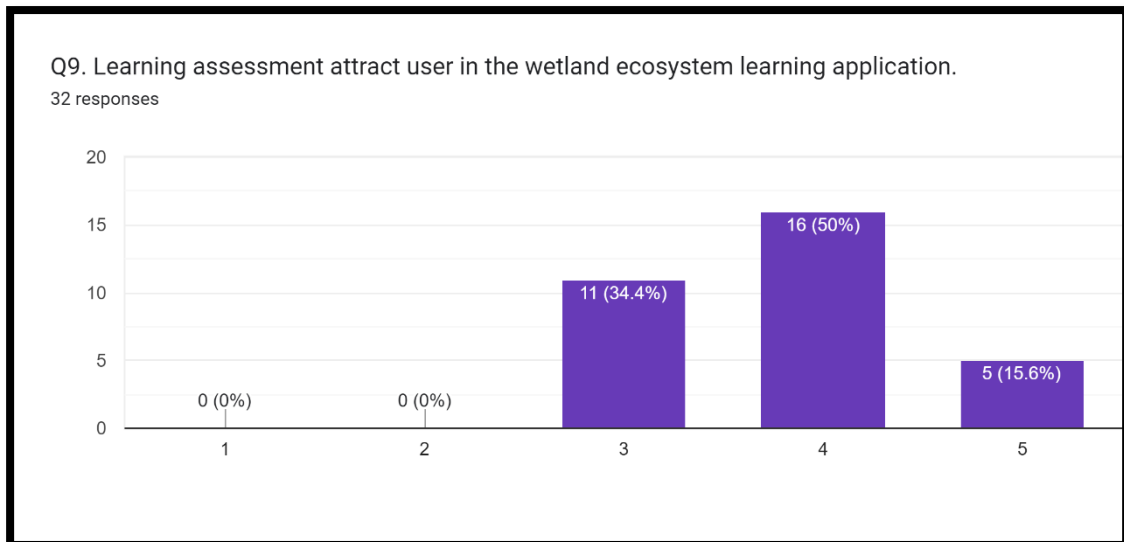
Figure 2.26 Column chart of Q12

The column chart above shows the number and percentage of Q12 from the 32 responses. Majority of them are agree that Theory-base is the most direct and effective way to increase knowledge in learning. 15 respondents agree and 5 respondents strongly agree with that total is 62.5%, 11 respondents stand neutral, and 1 respondent disagree theory-base is a direct way to increase knowledge in learning.
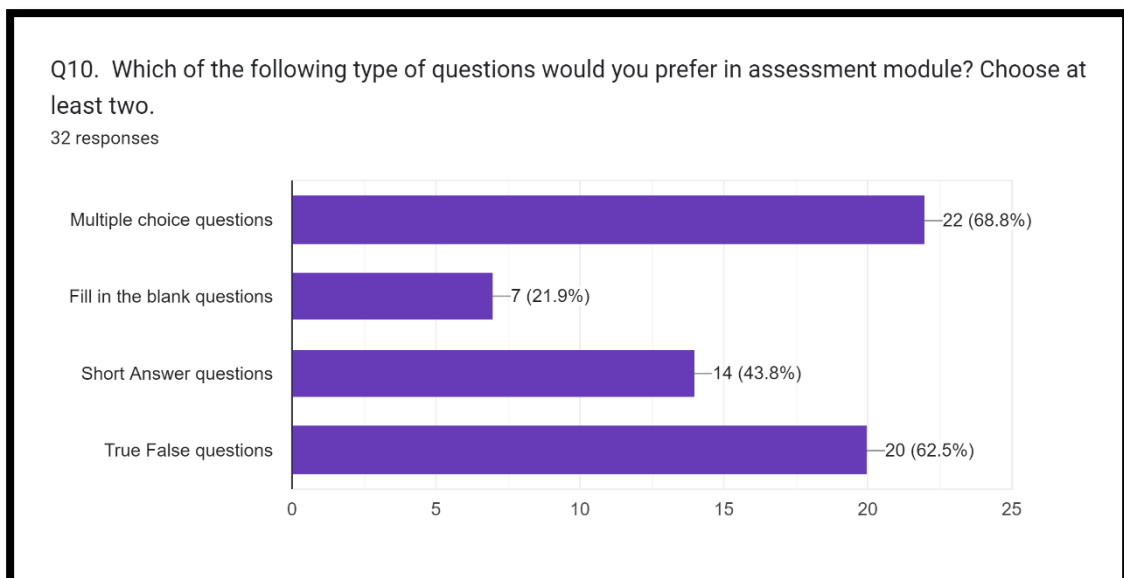

Figure 2.27 Bar chart of Q13

The bar chart above shown that most of the response of Q13 which of the following type of material would you prefer in theory-base module. The most of response for

the prefer following type of theory-base material in learning applications are video-base material. Video will include the text, graphic, image and audio together based on timeline, this is one of the learning ways users like to use in nowadays. The Large number of respondents prefer animation-base material and graphic-base material. Audio and text material only have 12 and 9 responses because these two type questions most of the young user already don't like to use anymore.
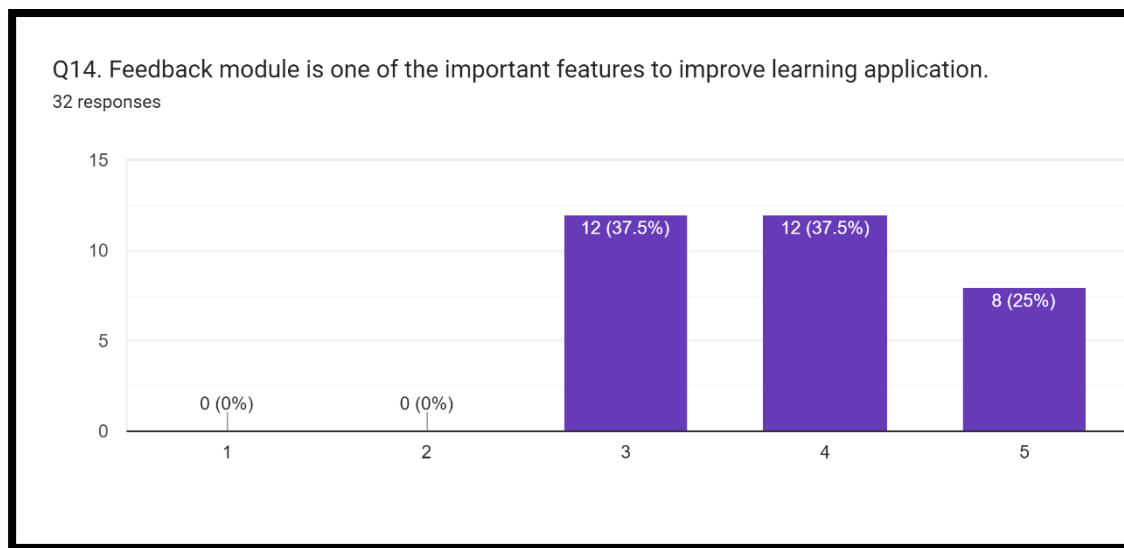
Figure 2.28 Column chart of Q14

 The column chart above shows the number and percentage of Q14 from the 32 responses. Majority of them are agree that feedback module is one of the important features to improve learning application. 12 respondents agree and 8 respondents strongly agree with that total is 62.5%, 12 respondents stand neutral, and no respondent disagree feedback can improve learning application.
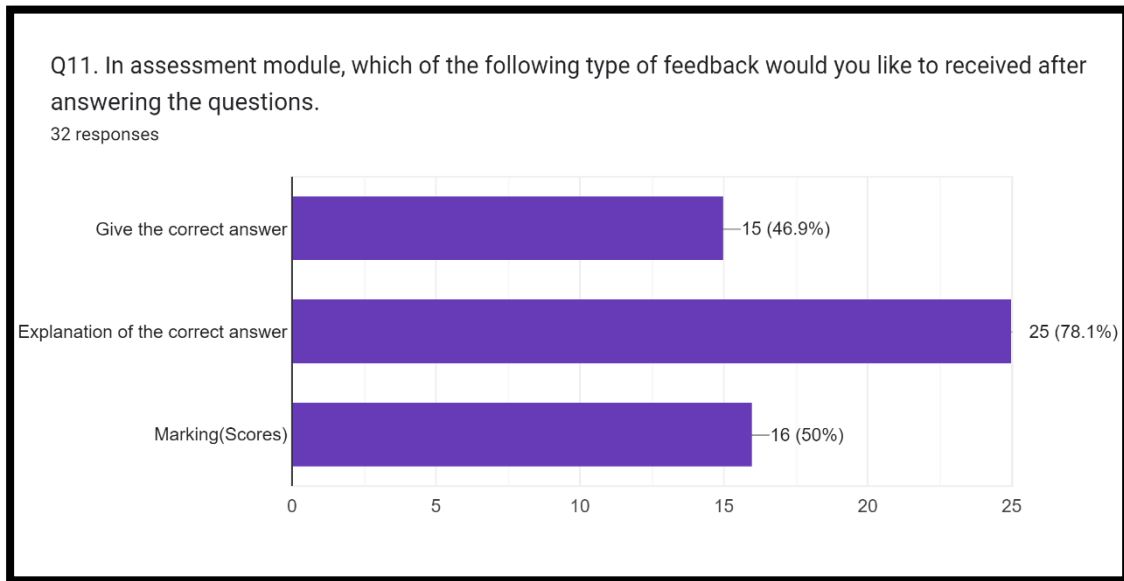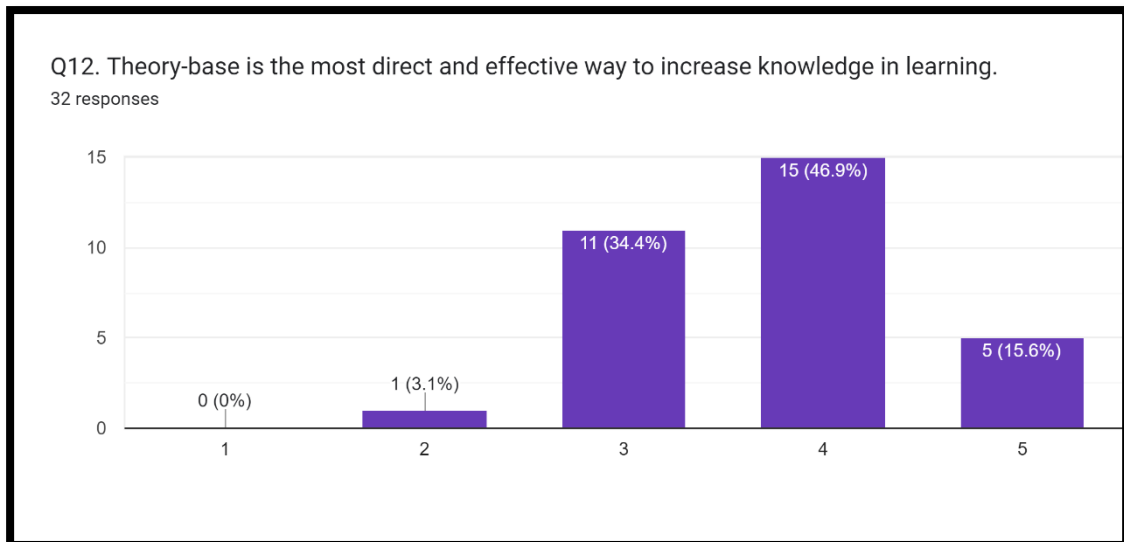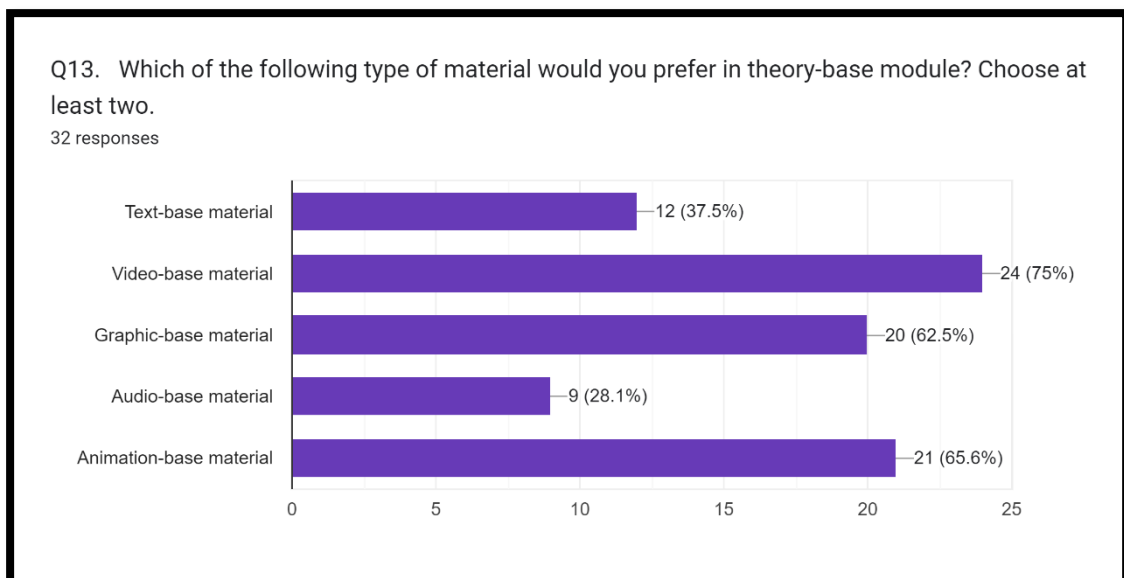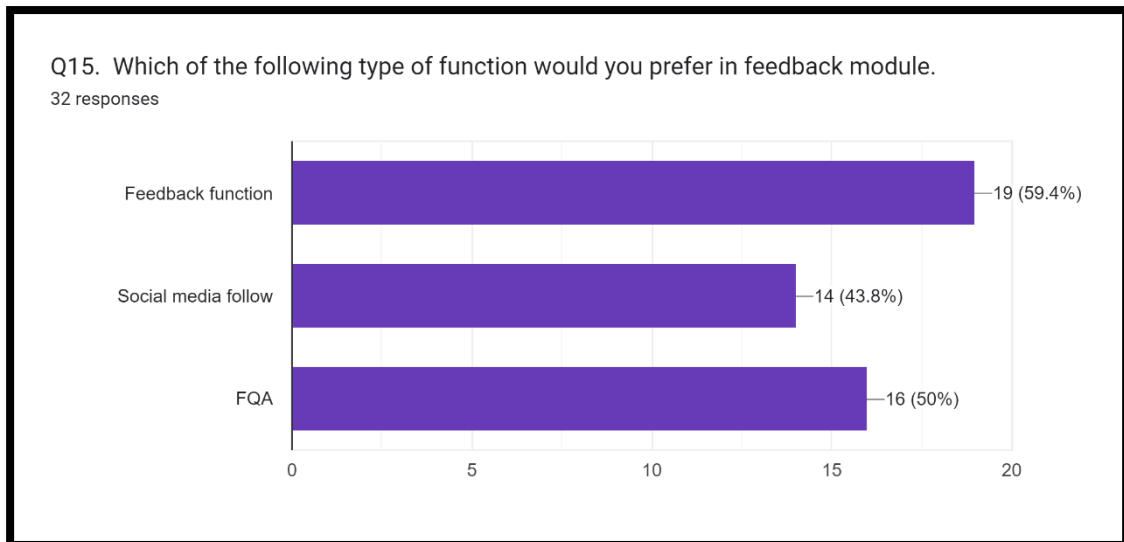
Figure 2.29 Bar chart of Q15

The bar chart above shown that most of the response of Q15 which of the following type of function would you prefer in feedback module. The most of response for the prefer type of feedback function in learning applications are feedback to us. Feedback is one of the important elements to continuously improve the app. FAQ and social media function have 16 and 14 respondents prefer to use these two functions. FAQ provide the frequently questions answer to user flexible to answer the user questions, but the answer may not be useful for every user. Social media follow may be is unnecessary to some of the user, they just to learn something.

2.6.3 Discussion

Based on the questionnaire results, it appears that the majority of respondents, particularly younger users, view learning applications as a valuable tool for acquiring new knowledge. The main reasons for this are the ease of use and user-friendly nature of such applications. However, respondents have also highlighted the issue of virtual learning environments (VLEs), with many feeling that current VLEs are not sufficient for their needs, and 3D VLEs are becoming increasingly important. It is worth noting that while respondents prefer user-friendly designs, they also do not want the modules to be too complex or have too many functions, as they only have limited time (around 1 hour) to use the application.

From the five module questions, it is clear that users want a simple and easy way to explore the study material, as the traditional text-based approach is viewed as too boring. The preferred functions in these modules have been identified, which will assist in the development phase of the application. Additionally, many respondents suggested that the application should incorporate AI and be user-friendly, which are both important factors in modern applications.

CHAPTER 3

METHODOLOGY

3.1 Overview

The software development methodology part of a software development project report outlines the processes and procedures that were followed throughout the software development life cycle (SDLC). The introduction to the section on software development process often provides an outline of the technique that was used on the project. The rationale for selecting the technique, its benefits, and how well it meets the project's objectives should all be covered in the introduction. The approach used may be Agile, such as Scrum or Kanban, or traditional, such as the Waterfall model. In general, each report on a software development project must include a section on software development techniques. It provides an exhaustive account of the development process, outlining all the strategies, steps, roles, tools, and technology used. Additionally, it provides a chance to evaluate the project's successes and identify areas that could use improvements in upcoming software projects.

Table 3.1 Comparison methodologies

Methodology	Type	Emphasis	Phases	Advantages	Disadvantages
ADDIE	Linear, sequential	Thorough analysis	Analysis, Design, Development, Implementation, Evaluation	Ensures that all relevant actions are done; ideal for projects with specific needs and learning objectives.	Often rigid for projects with shifting requirements; not always appropriate for all kinds of projects
XP	Agile	Simplicity, feedback, communication	Continuous planning, designing,	Increases software development efficiency and	Large, sophisticated projects might not be appropriate; a skilled

			coding, testing, listening	quality, lowers error risks, and is flexible to changing requirements	team is needed.
Scrum	Agile framework	Collaboration, communication, teamwork	Sprint planning, Daily Scrum, Sprint Review, Sprint Retrospective	Increases team flexibility and productivity, encourages ongoing improvement, and is flexible enough to meet changing needs.	Does not function well for projects with ambiguous criteria or goals because it requires a specialised Scrum Master.
Crystal	Family of Agile methodologies	Communication, teamwork, simplicity	Varies (e.g. Crystal Clear, Crystal Orange)	Good for small to medium-sized projects; places an emphasis on teamwork and communication; versatile and flexible	Unsuitable for complicated, large-scale projects; challenging to scale up
FDD	Agile	Creation of small, manageable features	Develop overall model, build feature list, Plan by feature, Design by feature, Build by feature	Focuses on design, testing, and project management; suitable for projects with precise needs and goals	Projects with shifting requirements might not be acceptable; careful planning and management are necessary

3.2 ADDIE Model
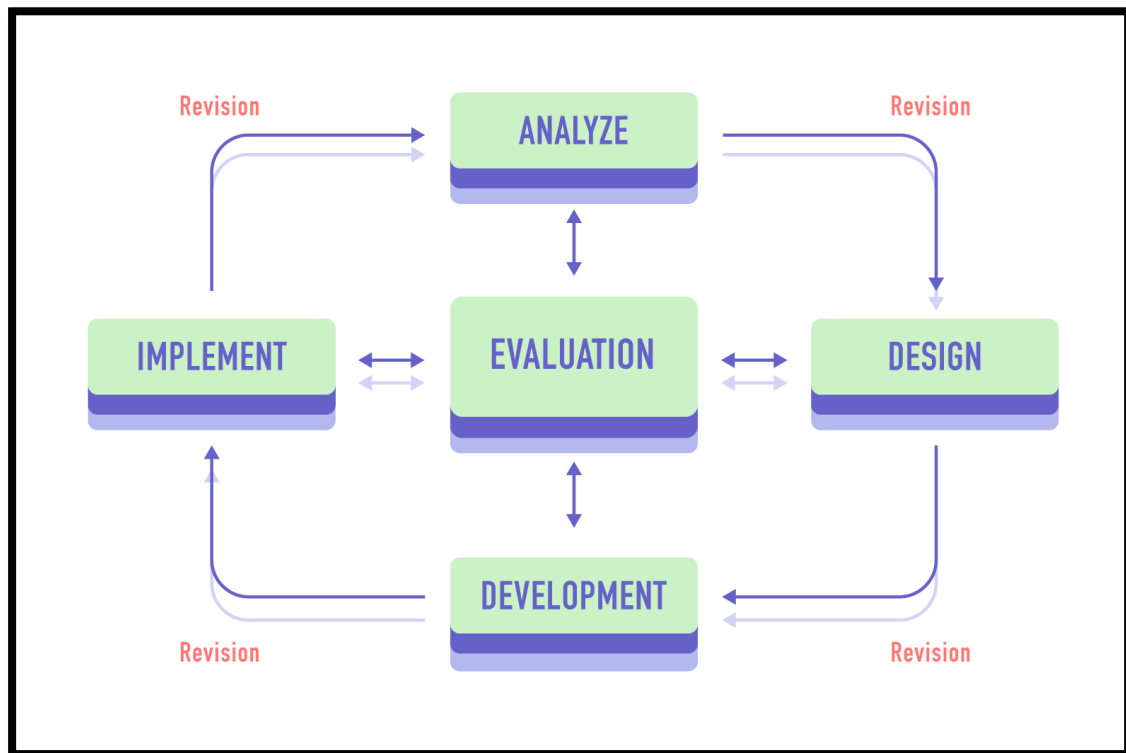
Figure 3.1 ADDIE Model

Source: (Eoghan Quigley, no date)

The ADDIE model is a popular instructional design framework for creating training materials and creating classes. The Center for Educational Technology at Florida State University developed the idea in 1975 for the American Army. The U.S. Armed Forces adopted the ADDIE training programme soon after it was created (Branson et al., 1975).

The ADDIE method states that the development of training support tools takes place in five stages or phases: analysis, design, development, implementation, and evaluation. By tailoring instruction to jobs and providing training in areas most essential to job performance, the process' initial objective was to increase the efficacy and efficiency of education and training (Allen, 2006).

3.2.1 Analysis phase

asserts that all other stages of instructional design, including the ADDIE process, are built upon the analysis phase. Potential educational issues and goals are identified at this preliminary level. In order to establish the kind and amount of instruction required, learners' current knowledge and skills are also assessed (Muruganantham Ganesan, 2015).

The analysis step can include particular study techniques, such as needs analysis, goal analysis, and task analysis, it is further noted. For instance, a needs analysis method will assist instructional designers in determining the resources needed and any possible limitations to their plans of action (Muruganantham Ganesan, 2015).

In this phase, first we should do some research and background study, identify the problem statement, define the objective and project scope. Second, we should do more research about literature review, questionnaire analysis and define target user.

3.2.2 Design phase

The method by which learners will accomplish the desired learning objectives is outlined by instructional designers during the design phase of the ADDIE model. A systematic approach should be used to carry out the design process, and a specific set of guidelines should be followed (Kurt, 2017).

Data gathered or acquired during the analysis phase is used as input for the design phase, assisting instructional designers in selecting teaching methods and resources that will be most successful for the learners involved(Arkün and Akkoyunlu, 2008). At this point in the ADDIE model, timetables for learning exercises and feedback systems are also established. This consists of a plan, delivery techniques, structure, timeframe, evaluation, and feedback (Imed Bouchrika, 2022).

In the design phase, based on the previous analysis and strategy should storyboard your ideas and create a prototype. A storyboard can visualize your ideas in an easy way and prepare them for production. You are creating a blueprint for your project,

and by making a prototype you can quickly communicate with other stakeholders the value of the training (Eoghan Quigley, no date).

3.2.3 Development phase

You can start building the routes at this point. At this stage, the prototype and storyboards will serve as your primary guides. Every component of the course should be created to correspond with the design process. Exercises, lectures, simulations, and other training resources are all included in the newly created content (Mayfield, 2011).

The frustrating issue is that because the instructional designer did not test the course, they frequently miss something very simple. Iterative growth should be used. Once a module has been made, it should be tested to make sure there are no fundamental mistakes in grammar, spelling, syntax, etc. Testing the course involves systematically evaluating the accuracy of the material and the usefulness of the navigation, not simply skimming through it (Eoghan Quigley, no date; Imed Bouchrika, 2022).

In this phase, the author should start to develop our application, create the 3D modeling using Blender, build the 3D VLEs using Unity and testing.

3.2.4 Implementation phase

It is time to share your courses with the learner once you have finished them and are confident that they have been thoroughly tried. How this is actually carried out will be influenced by the choices made during the design process. Most of the time, the courses are uploaded to an LMS, and the delivery options are made up, including who is enrolled, how much time is allotted, the pass rates for assessments, and the gathering of comments (Eoghan Quigley, no date). The LMS manages the delivery, monitoring, and reporting.

In the implementation phase, the learning application is made available to the target users. During this phase, the author will strive to make sure the application is completely functional and that it satisfies all user needs to make sure the application is completely functional and that it satisfies all user needs. In order to gather feedback from the target users and gauge their degree of satisfaction with the application, the team will also carry out a thorough user survey.

3.2.5 Evaluation phase

The evaluation part assesses the instructional program's efficacy and efficiency. Assessment is at the core of the procedure in the updated ADDIE model (Allen, 2006). The analysis part is where the evaluation process starts, and it lasts the entire duration of the learning program.

Formative and summative assessments are both a part of the evaluation process. At each step of the ADDIE process, products and procedures are evaluated in formative assessments to guarantee quality and ongoing improvement (Imed Bouchrika, 2022). a comprehensive summative evaluation of the study program's results, including a determination of the program's overall efficacy (Allen, 2006). The procedures and results used in each step of the ADDIE training model are assessed in relation to the instructional requirements and learning principles as instructional designers progress through the various stages. It's crucial to get input on all parts of the course so that you can revise and improve the material and identify any gaps you may have missed (Imed Bouchrika, 2022).

In the evaluation phase, the team will analyze the feedback collected from the users and assess the overall effectiveness of the learning application. The team will use this feedback to adjust or enhance the application as necessary to meet the demands of the target users and achieve the desired learning objectives.

3.3 System Requirements

3.3.1 Software Requirements

Table 3.2 Software Requirements

Elements	Type of software	Description
Text	Word	Used for writing the report, cutting, copying and pasting of text elements.
Graphic	Adobe Illustrator / vectr/ Figma	Used to edit graphics such as animal, background and pyramid.
Image	Adobe Photoshop/ Photopea	Used to edit image such as animal, background and pyramid.
Video	Cap cut/ Adobe Premium Pro	Use to create video such as introduction video and instruction video.
2D Animation	Adobe Animate	Use to create 2D modeling and animation.
3D Animation	Blender/Maya	Use to create 3D modeling and animation.
Authoring Tools	Unity / Adobe Director	Use to develop cross platform multimedia presentation, animation and interaction application.

3.3.2 Hardware Requirements

Table 3.3 Hardware Requirements

Hardware	Minimum Requirement	Optimal Requirement
OS	Window 8	Window 11
Processor	AMD RYZEN 5	AMD RYZEN 7
CPU	2.5GHz	3.4GHz
RAM	4 GB	16 GB
Storage	256 GB	1 TB

Graphic Card	Nvidia GeForce	Nvidia GeForce GTX

3.3.3 Development platforms and tools

3.3.3.1 Blender

Blender is a 3D computer graphics software that is used for creating 3D models, animations, and visual effects. It is an application of free and open-source software that runs on Windows, Mac, and Linux. A wide range of modelling tools, including sculpting, mesh editing, and texturing, are available in Blender. Additionally, it contains a strong animation system that enables users to build elaborate simulations and animations. Blender provides tools for visual effects production, compositing, and rendering in addition to modelling and animation. It has a robust rendering engine that can create stunning images and animations, a node-based compositing system that enables users to combine many layers and effects, and more.

3.3.3.2 Unity

Unity is a popular game engine and development platform used for creating 2D and 3D games and other interactive experiences. Unity is compatible with mobile operating systems like iOS and Android as well as Windows, Mac, and Linux. A graphic editor, physics engine, animation tools, scripting support, and a large selection of assets and plugins that are accessible in the Unity Asset Store are just a few of the features and tools that Unity offers for game production. Unity's user-friendliness and accessibility are two of its main advantages. It offers a user-friendly interface that enables programmers to make interactive experiences and games without having to write a lot of code. In addition, it offers developers strong scripting capabilities so they may modify and customize their projects.

3.4 System Design

3.4.1 System Flow Diagram
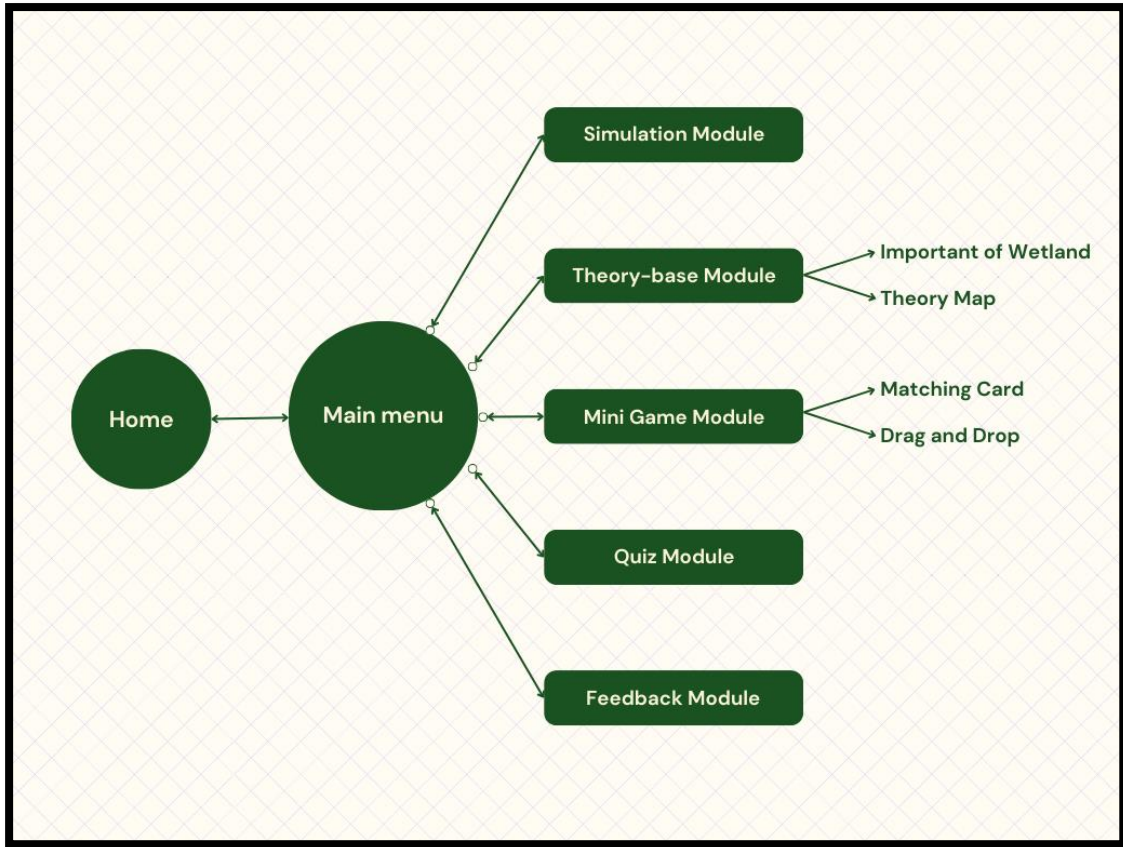
Figure 3.2 System Flow Diagram

3.4.2 Storyboards Design

Topic: Home

Storyboard no: 1

Figure 3.3 Home storyboard

Table 3.4 Home description and flow diagram

Description	Flow Diagram
T1: show the title of the application. **B1:** show the start button navigate to the main menu. **G1:** show the home page background.	Home Start Display T1, B1, G1 Click B1 Display the questions answers End

Topic: Main Menu

Storyboard no: 2
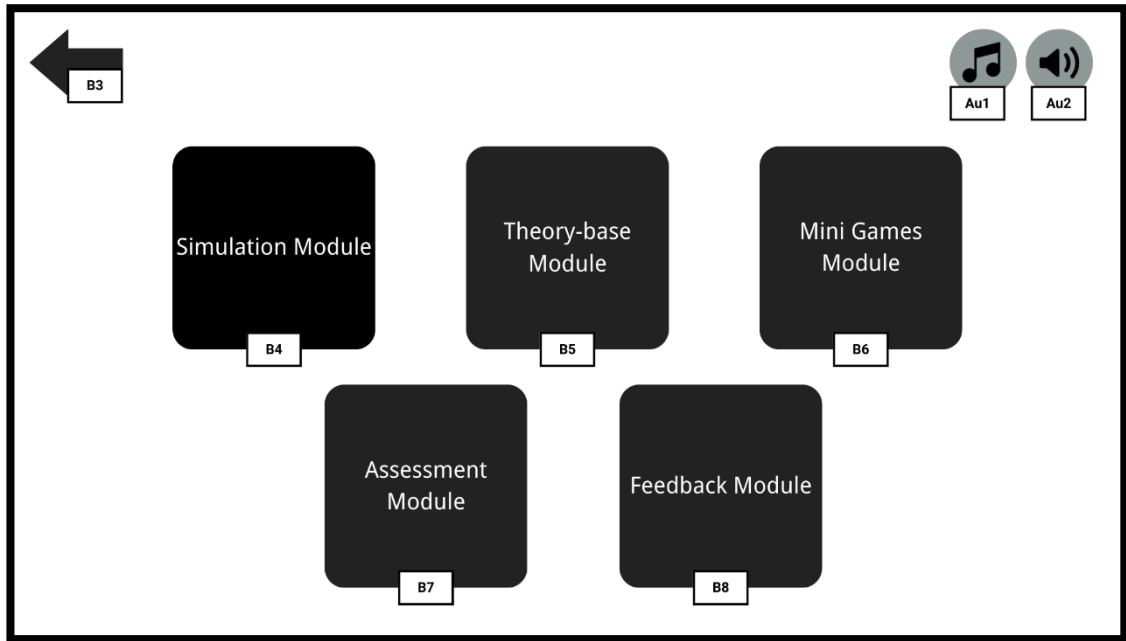
Figure 3.4 Main menu storyboard

Table 3.5 Main menu description and flow diagram

Description
Au1: show the music icon to turn on/off background music.
Au2: show the audio icon to turn on/off the audio sound.
B3: show the back button navigate to the home page.
B4: show the Simulation button navigate to Simulation Module page.
B5: show the Theory-base button navigate to Theory-base page.
B6: show the Mini Games button navigate to Mini games Module page.
B7: show the Assessment button navigate to Assessment Module page.
B8: show the Feedback button navigate to Feedback Module page.
Flow Diagram

Main Menu
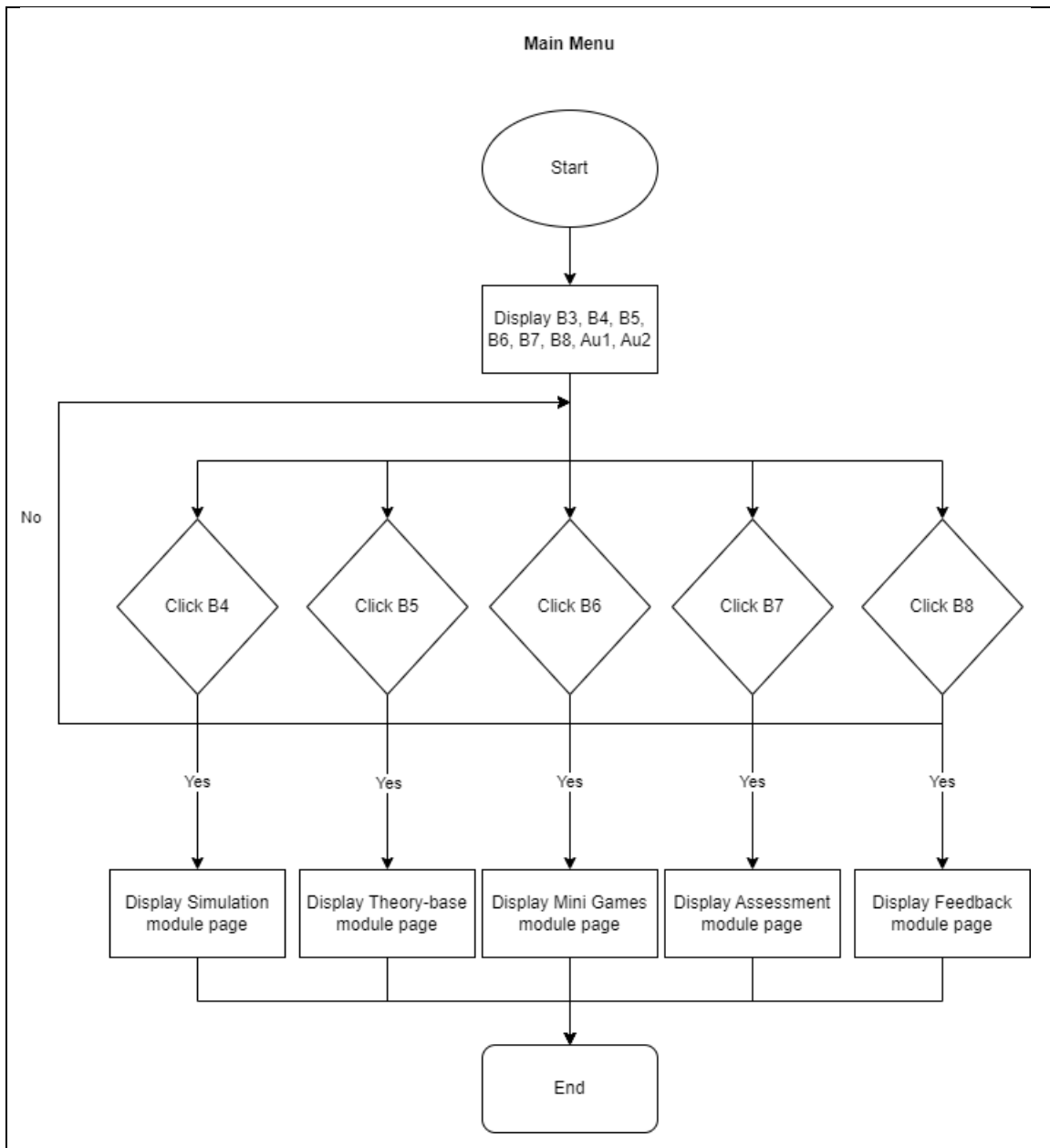
Topic: Simulation Module

Storyboard no: 3
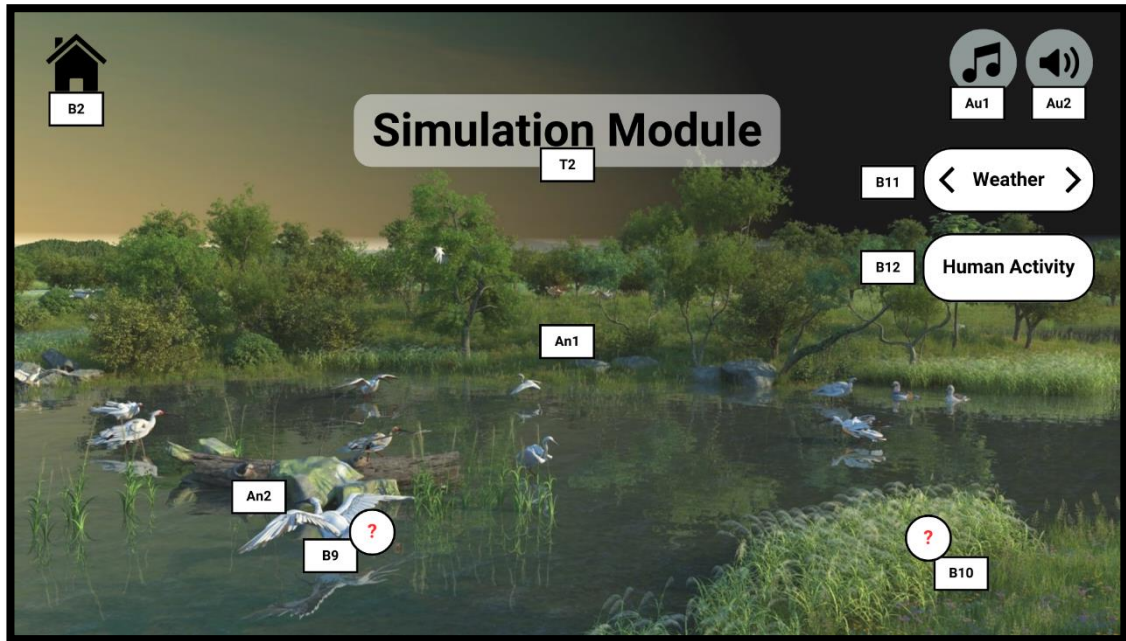
Figure 3.5 Simulation page storyboard

Table 3.6 Simulation description and flow diagram

Description	Flow Diagram
B2: show the home button navigate to main menu page. **Au1:** show the music icon to turn on/off background music. **Au2:** show the audio icon to turn on/off the audio sound. **T2:** show the module tile. **B9:** show the button navigate to the animal's information. **B10:** show the button navigate to the plant's information. **B11:** show the button to change the weather. **B12:** show the button to action the human activity. **An1:** show the 3D wetland environment. **An2:** show the 3D animal modeling.	

Topic: Theory-base Module

Storyboard no: 4
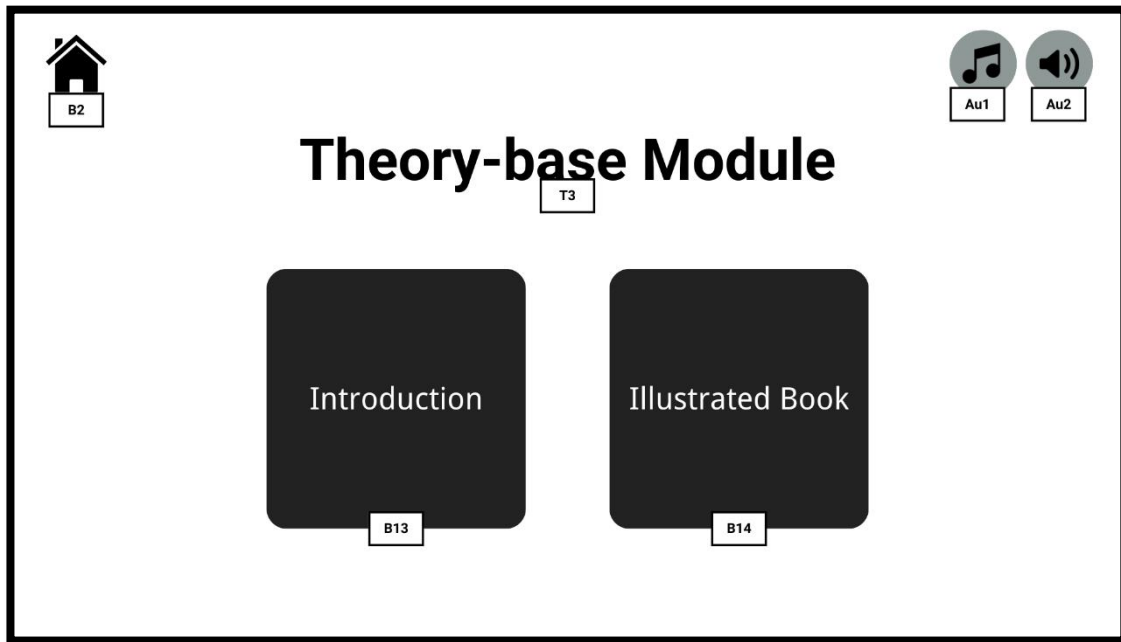
Figure 3.6 Theory-base page storyboard

Table 3.7 Theory-base description and flow diagram

Description	Flow Diagram
B2: show the home button navigate to main menu page. **Au1:** show the music icon to turn on/off background music. **Au2:** show the audio icon to turn on/off the audio sound. **T3:** show the module title. **B13:** show the button navigate to Introduction page. **B14:** show the button navigate to Illustrated book page.	

Topic: Introduction page

Storyboard no: 5
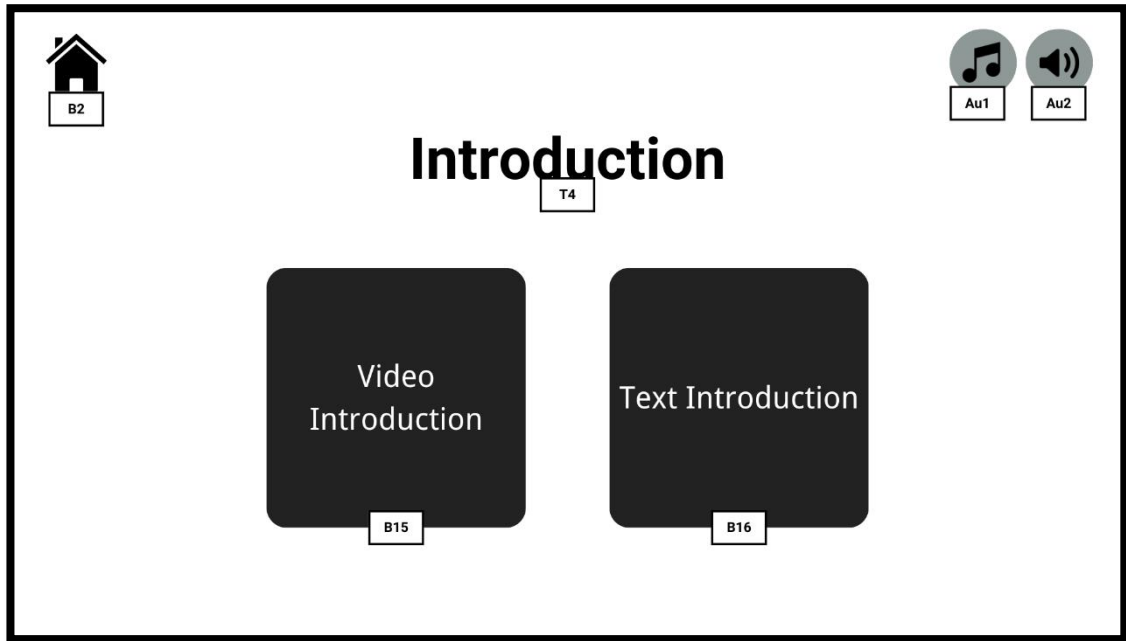
Figure 3.7 Introduction storyboard

Table 3.8 Introduction description and flow diagram

Description	Flow Diagram
B2: show the home button navigate to main menu page. **Au1:** show the music icon to turn on/off background music. **Au2:** show the audio icon to turn on/off the audio sound. **T4:** show the sub module title. **B15:** show the button navigate to Video Introduction page. **B16:** show the button navigate to Text Introduction page.	

Topic: Video Introduction page

Storyboard no: 6
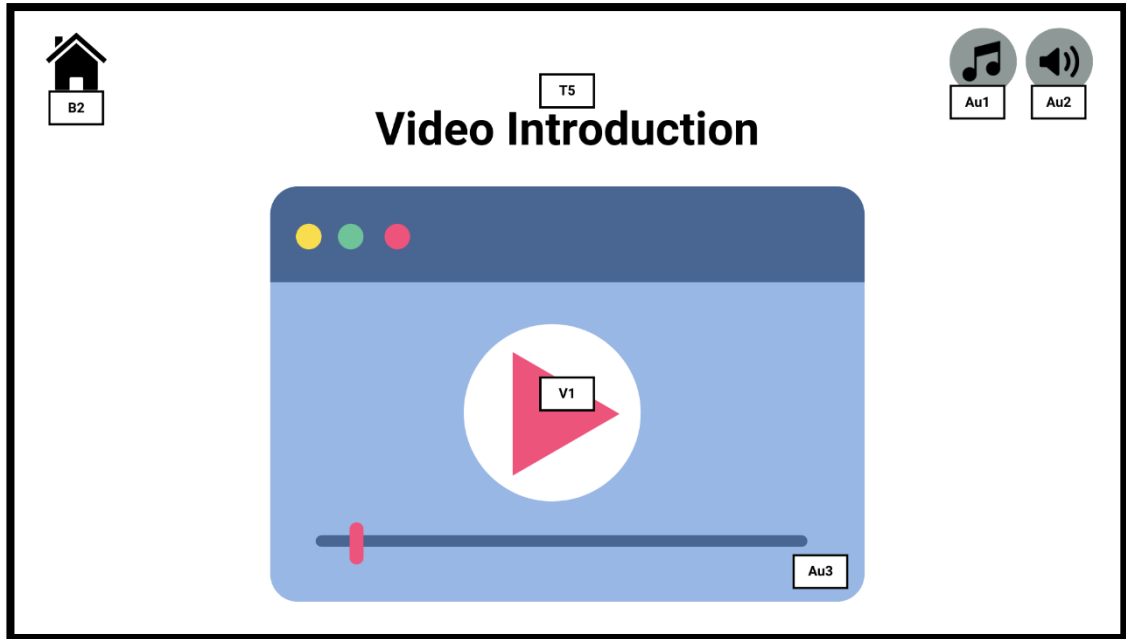
Figure 3.8 Video Introduction storyboard

Table 3.9 Video introduction description and flow diagram

Description	Flow Diagram

B2: show the home button navigate to main menu page. **Au1:** show the music icon to turn on/off background music. **Au2:** show the audio icon to turn on/off the audio sound. **T5:** show the module function title. **V1:** show the wetland introduction video. **Au3:** show the audio icon to mute and adjust audio level.	Video Introduction Start Display B2, Au1, Au2, V1, Au3 Click V1 — Yes — Display video to start Click Au3 — Yes — Display the audio on/off End

Topic: Text Introduction page

Storyboard no: 7
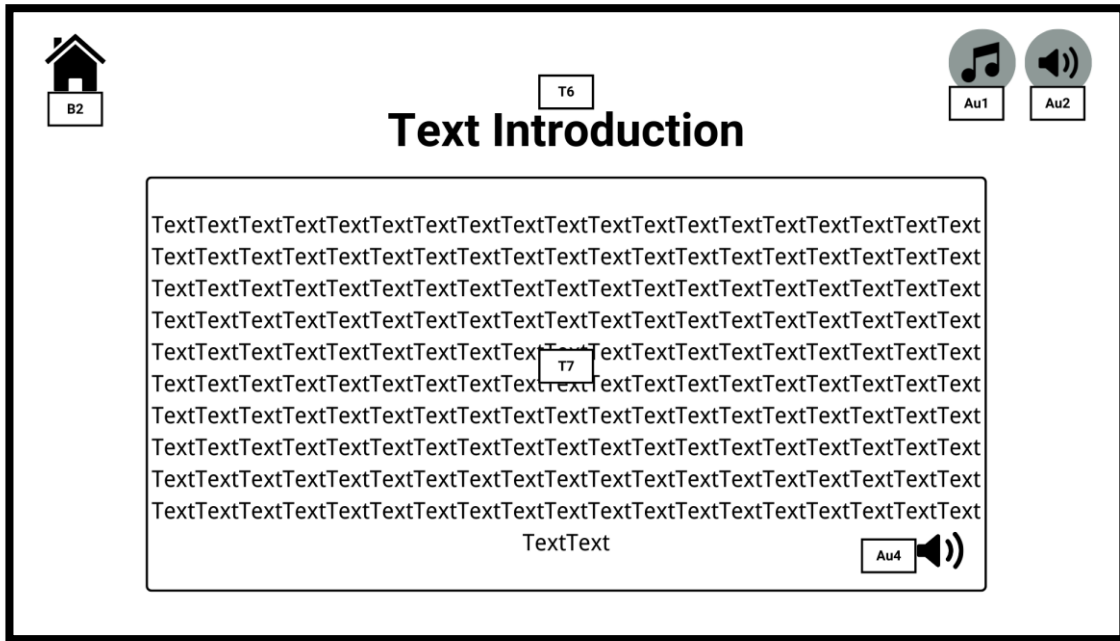
Figure 3.9 Text Introduction storyboard

Table 3.10 Text Introduction description and flow diagram

Description	Flow Diagram
B2: show the home button navigate to main menu page. **Au1:** show the music icon to turn on/off background music. **Au2:** show the audio icon to turn on/off the audio sound. **T6:** show the module function title. **T7:** show the wetland text introduction. **Au4:** show the audio icon to read out the text.	

Topic: Illustrated Book page

Storyboard no: 8
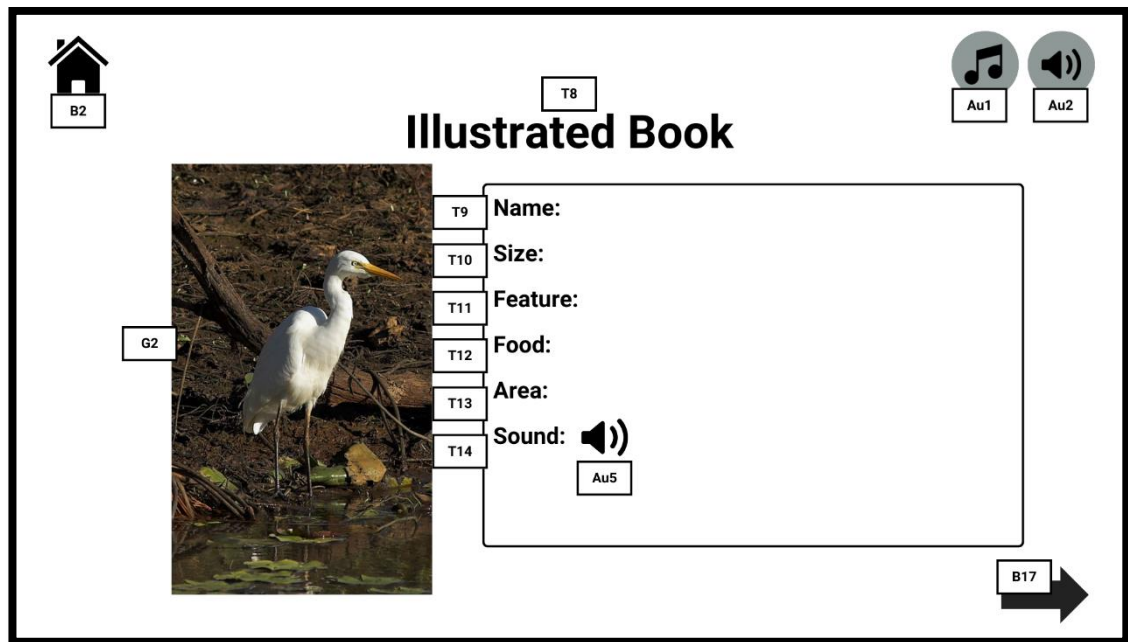
Figure 3.10 Illustrated book storyboard

Table 3.11 Illustrated book description and flow diagram

Description	Flow Diagram

B2: show the home button navigate to main menu page.

Au1: show the music icon to turn on/off background music.

Au2: show the audio icon to turn on/off the audio sound.

G2: show the animal/plant image.

T8: show the module function title.

T9: show the animal/plant name.

T10: show the animal/plant size.

T11: show the animal/plant feature.

T12: show the animal/plant food.

T13: show the animal/plant area.

T14: show the animal sound.

Au5: show the audio icon to play the animal sound.

B17: show the button navigate to the next animal page.

Illustrated Book

Start

Display T8, T9, T10, T11, T12, T13, T14, G2, Au5, B17

Click Au5

Click B17

Yes

Yes

Display to play animal sounds audio

Display to the next animal

End

Topic: Mini Games Module

Storyboard no: 9
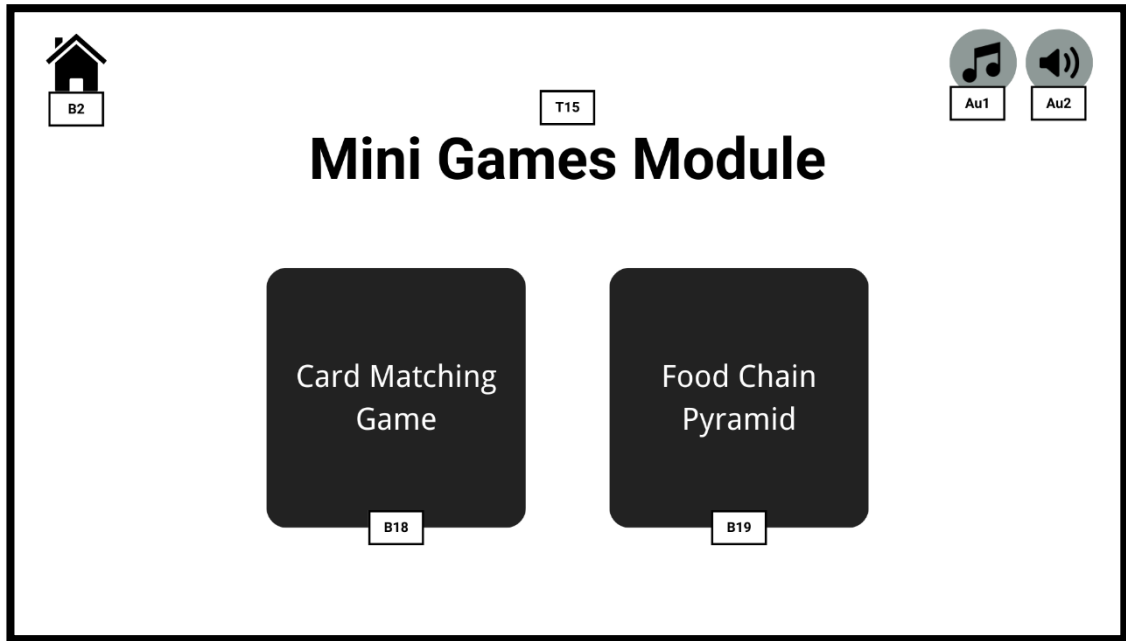
Figure 3.11 Mini games storyboard

Table 3.12 Mini games description and flow diagram

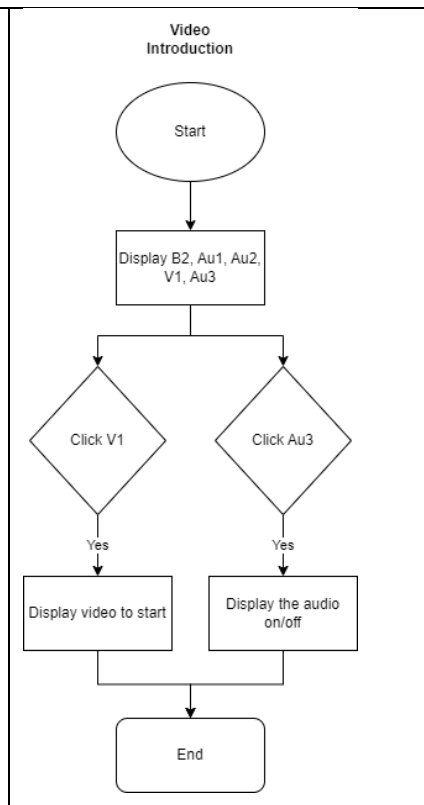| Description	Flow Diagram
B2: show the home button navigate to main menu page. **Au1:** show the music icon to turn on/off background music. **Au2:** show the audio icon to turn on/off the audio sound. **T15:** show the module title. **B18:** show the button navigate to Card Matching Game page. **B19:** show the button navigate to Food Chain Pyramid page.	

Topic: Card Matching Game

Storyboard no: 10
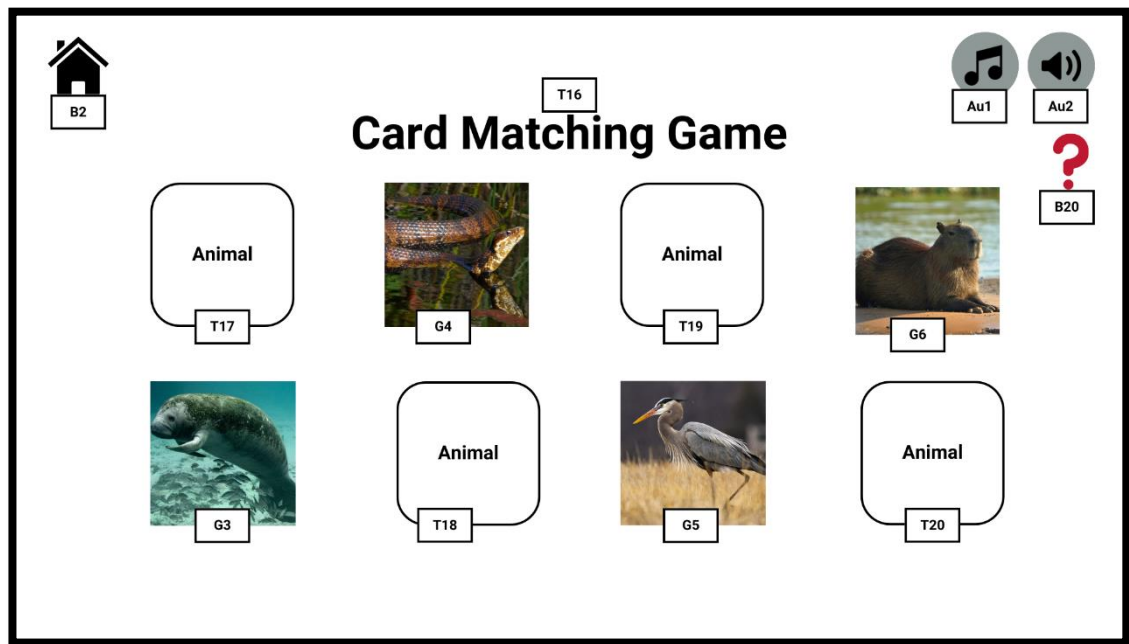
Figure 3.12 Card matching game storyboard

Table 3.13 Card matching description and flow diagram

Description	Flow Diagram

B2: show the home button navigate to main menu page.

Au1: show the music icon to turn on/off background music.

Au2: show the audio icon to turn on/off the audio sound.

T16: show the module function name.

T17: show the animal/plant name.

T18: show the animal/plant name.

T19: show the animal/plant name.

T20: show the animal/plant name.

G3: show the animal/plant image.

G4: show the animal/plant image.

G5: show the animal/plant image.

G6: show the animal/plant image.

B20: show the button to see the instruction how to play.

Card Matching Game

Start

Display B2, Au1, Au2, T16, T17, T18, T19, T20, G3, G4, G6, G6, B20

Click on the first T17, T18, T19, T20

No

Click on the second G3, G4, G6, G6

Yes

Eliminate this two card

if card number==0

No

Yes

Click on instruction button

Yes

Display the instruction information

End
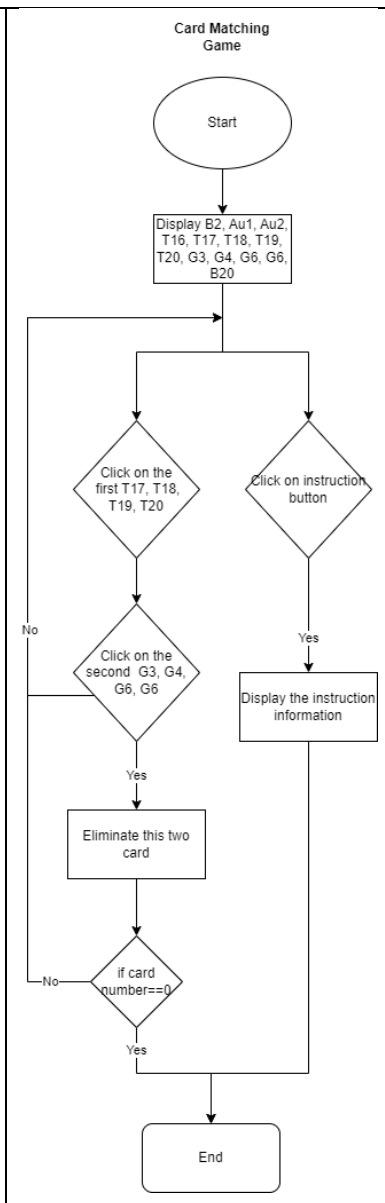
Topic: Food Chain Pyramid

Storyboard no: 11
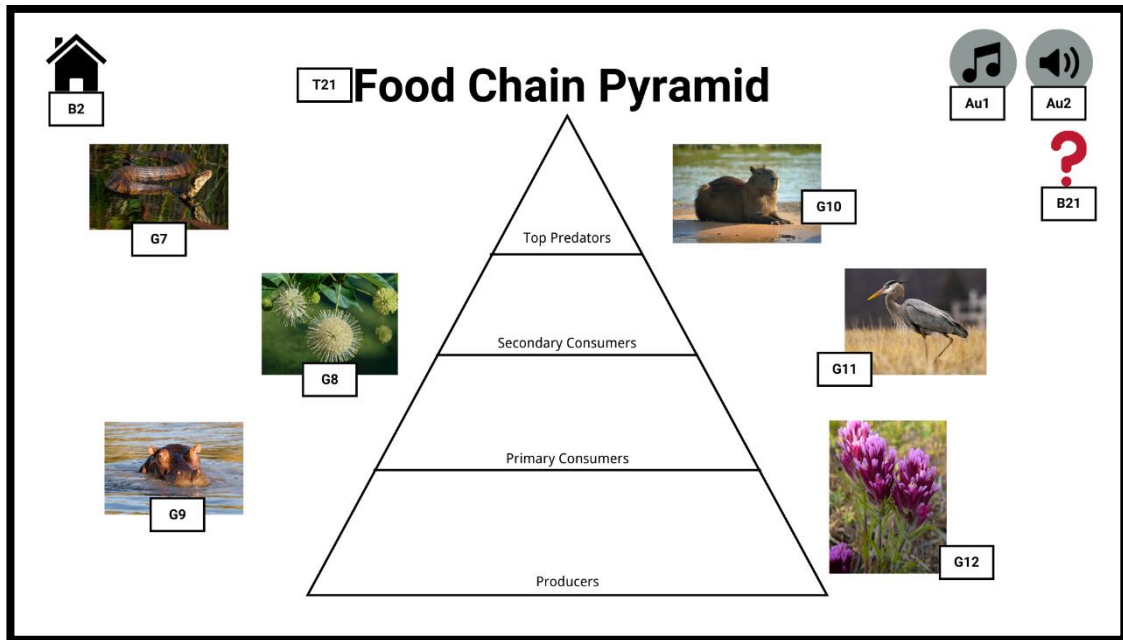
Figure 3.13 Food pyramid storyboard

Table 3.14 Food pyramid description and flow diagram

Description	Flow Diagram

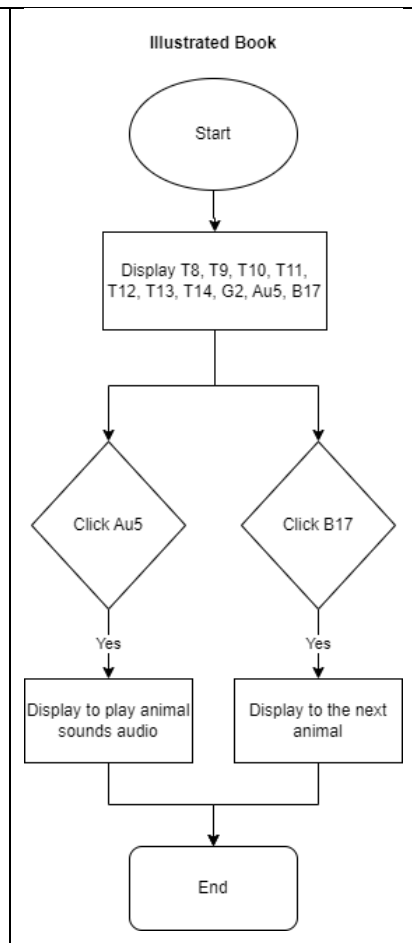|	
B2: show the home button navigate to main menu page. **Au1:** show the music icon to turn on/off background music. **Au2:** show the audio icon to turn on/off the audio sound. **T21:** show the module function title. **G7:** show the animal/plant image. **G8:** show the animal/plant image. **G9:** show the animal/plant image. **G10:** show the animal/plant image. **G11:** show the animal/plant image. **G12:** show the animal/plant image. **B21:** show the button to see the instruction how to play.	Food Pyramid Game Start DisplayB2, Au1, Au2, T21, G7, G8, G9, G10, G11, G12 No — Click any G7, G8, G9, G10, G11, G12 drag into pyramid Yes Display the correct message, Graphic number-1 Click on instruction button Yes Display the instruction information if Graphic number==0 — No Yes End

Topic: Assessment Module
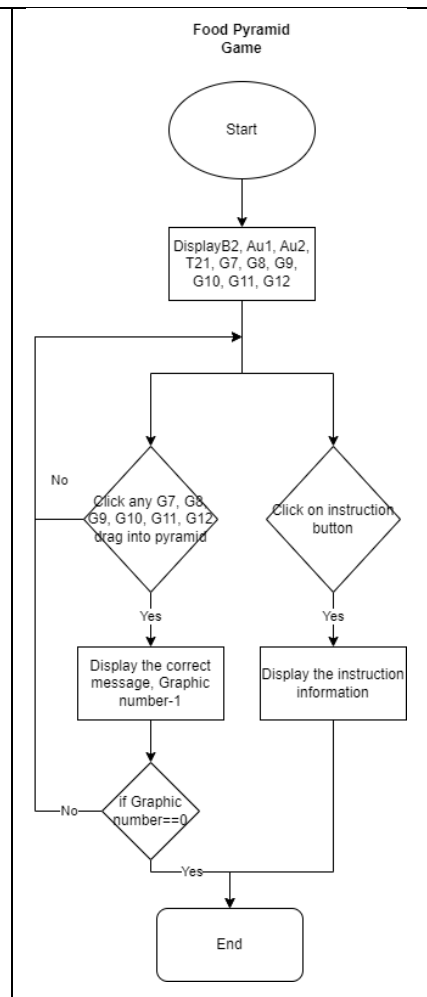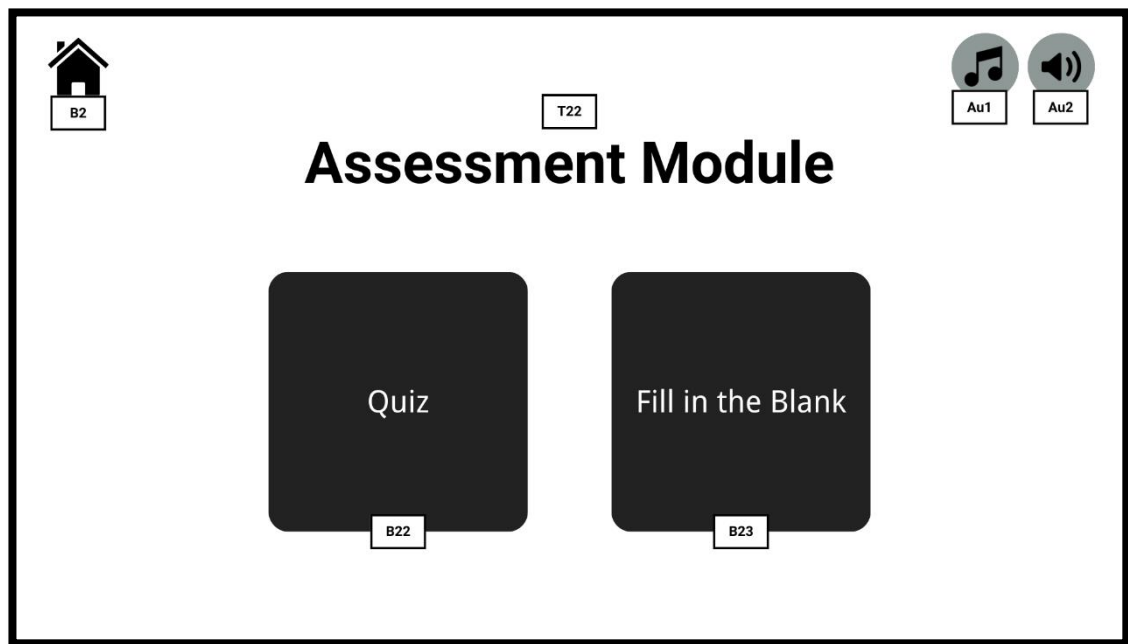
Storyboard no:12

Figure 3.14 Assessment storyboard

Table 3.15 Assessment description and flow diagram

Description	
B2: show the home button navigate to main menu page. **Au1:** show the music icon to turn on/off background music. **Au2:** show the audio icon to turn on/off the audio sound. **T22:** show the module title. **B22:** show the button navigate to Quiz page. **B23:** show the button navigate to Fill in the blank page.	

Topic: Quiz page

Storyboard no: 13
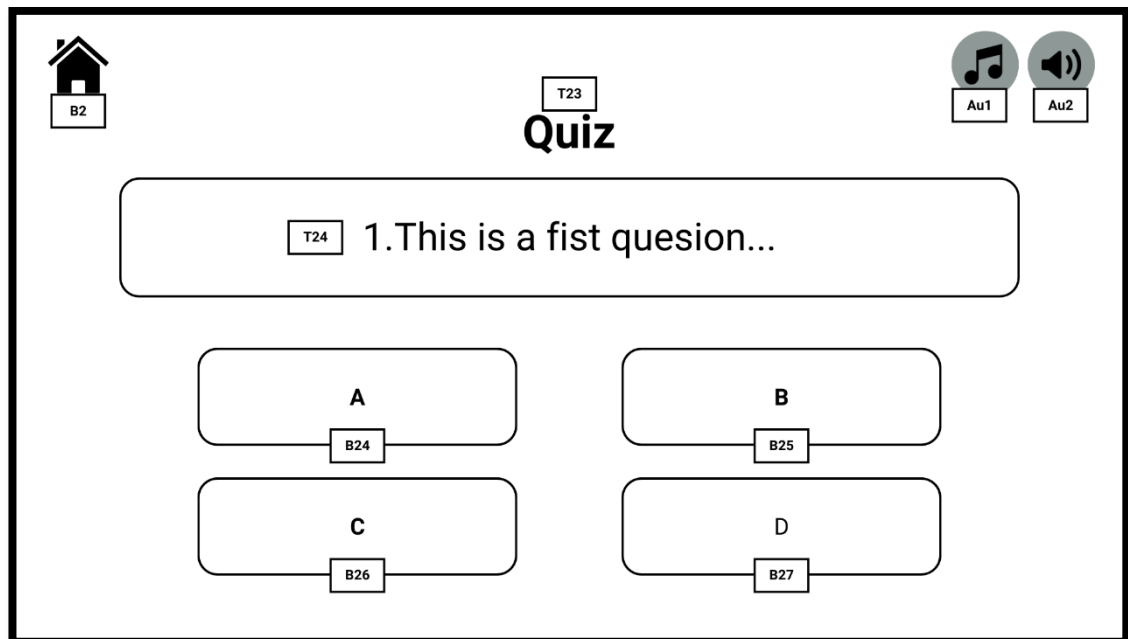
Figure 3.15 Quiz storyboard

Table 3.16 Quiz description and flow diagram

Description	Flow Diagram

B2: show the home button navigate to main menu page. **Au1:** show the music icon to turn on/off background music. **Au2:** show the audio icon to turn on/off the audio sound. **T23:** show the module function title. **T24:** show text for the question. **B24:** show the button used for option A. **B25:** show the button used for option B. **B26:** show the button used for option C. **B27:** show the button used for option D.	Quiz Start Display B2, Au1, Au2, T23, T24, B24, B25, B26, B27 No Click on the button Yes Display the correct message and next questions End
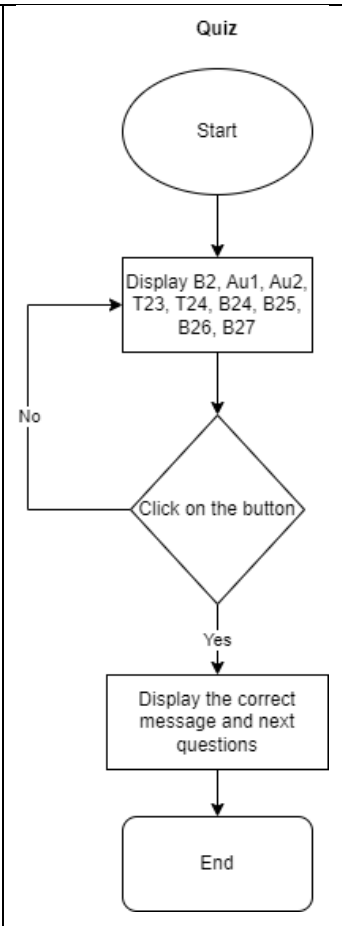
Topic: Fill in the blank

Storyboard no: 14
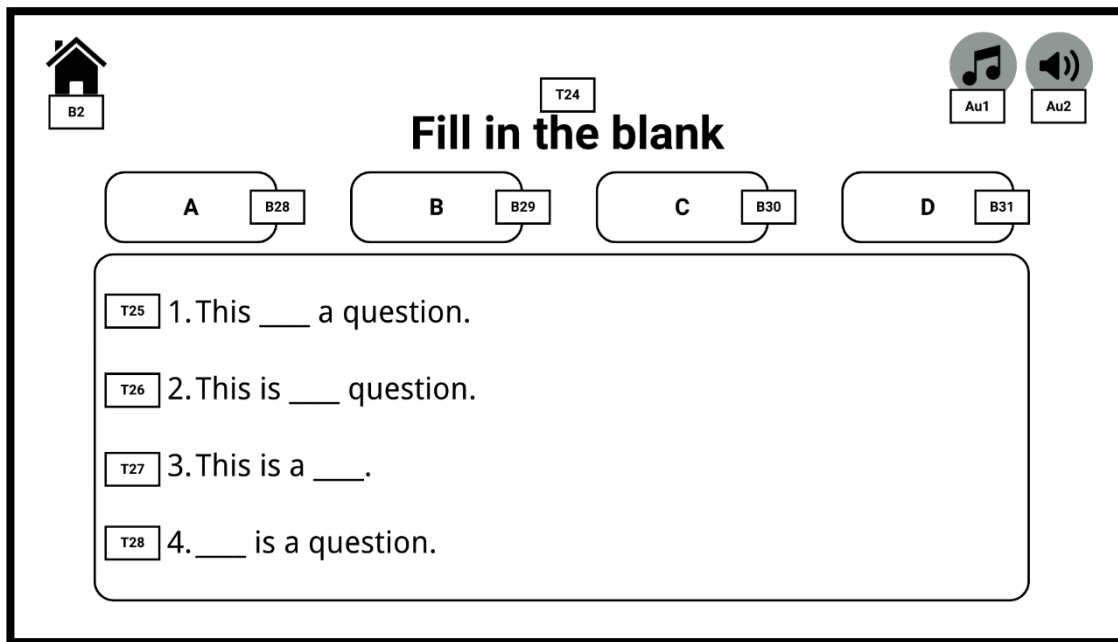
Figure 3.16 Fill in the blank storyboard

Table 3.17 Fill in the blank description and flow diagram

Description	Flow Diagram

B2: show the home button navigate to main menu page.

Au1: show the music icon to turn on/off background music.

Au2: show the audio icon to turn on/off the audio sound.

B28: show the button used for option A.

B29: show the button used for option B.

B30: show the button used for option C.

B31: show the button used for option D.

T24: show the module function title.

T25: show text for the question 1.

T26: show text for the question 2.

T27: show text for the question 3.

T28: show text for the question 4.

Fill in the Blank

Start

Display B2, Au1, Au2, B28, B29, B30, B31, T25, T26, T27, T28

Click on the choice and drag to the questions —No

Yes

Display the correct message, choice number-1

if Text choice number==0 —No

Display the all correct message

End

Topic: Feedback Module
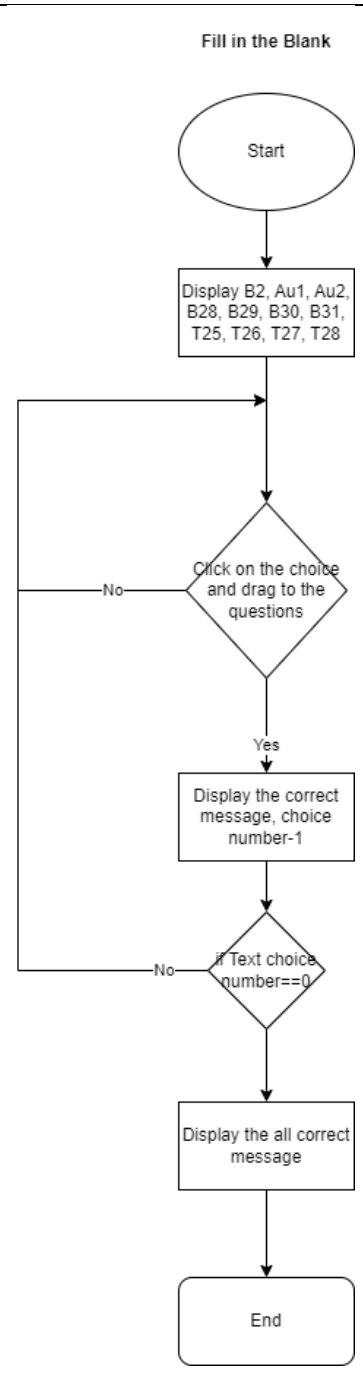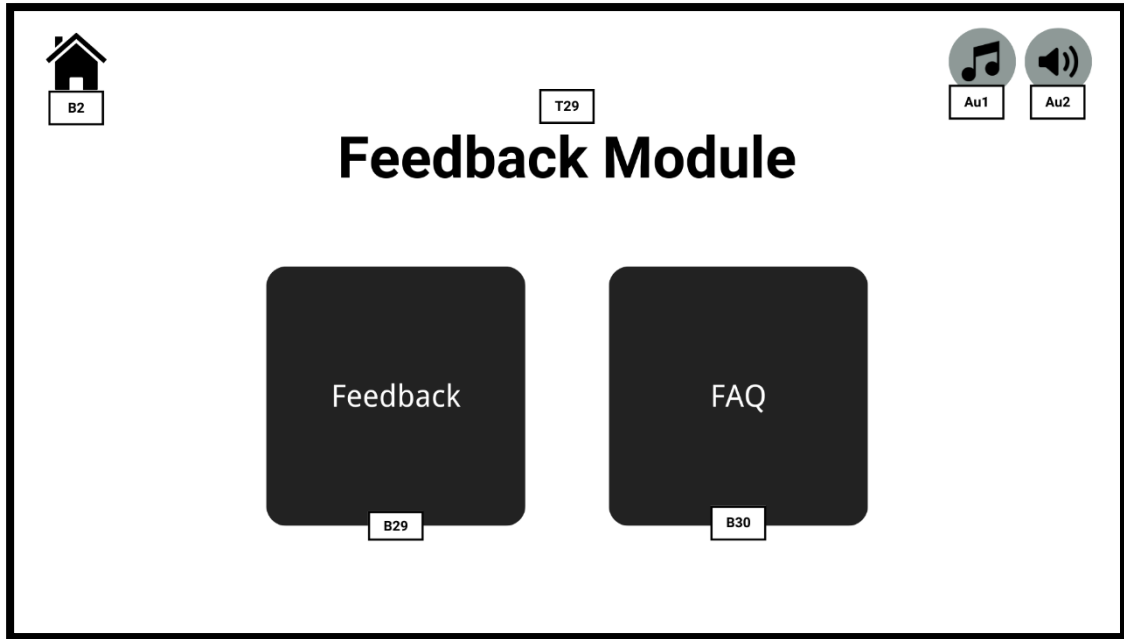
Storyboard no: 15

Figure 3.17 Feedback storyboard

Table 3.18 Feedback description and flow diagram

Description	Flow Diagram
B2: show the home button navigate to main menu page. **Au1:** show the music icon to turn on/off background music. **Au2:** show the audio icon to turn on/off the audio sound. **T29:** show the module title. **B29:** show the button navigate to the feedback page. **B30:** show the button navigate to the FAQ page.	

Topic: Feedback page

Storyboard no: 16
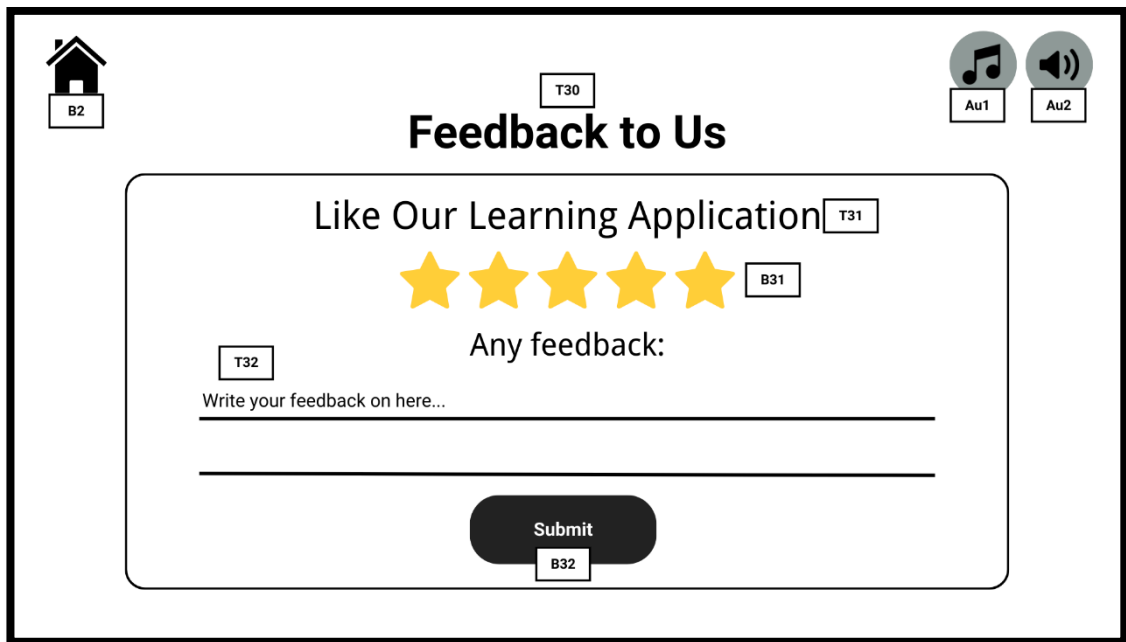
Figure 3.18 Feedback to us storyboard

Table 3.19 Feedback to us description and flow diagram

Description	Flow Diagram

B2: show the home button navigate to main menu page. **Au1:** show the music icon to turn on/off background music. **Au2:** show the audio icon to turn on/off the audio sound. **T30:** show the module function title. **T31:** show the text for feedback questions. **T32:** show the text for user write their feedback. **B31:** show the button to give the comment rating. **B31:** show the button to submit user feedback.	Feedback (Start) ↓ [Display B2, Au1, Au2, T30, T31, T32, B31, B32] ↓ < Click B31, write out T32 and click B32 > ↓ (End)
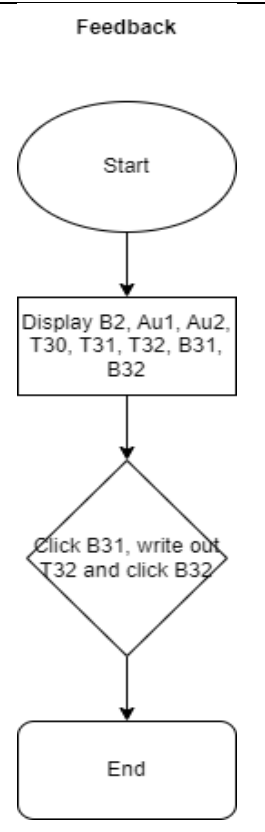
Topic: FAQ page

Storyboard no: 17
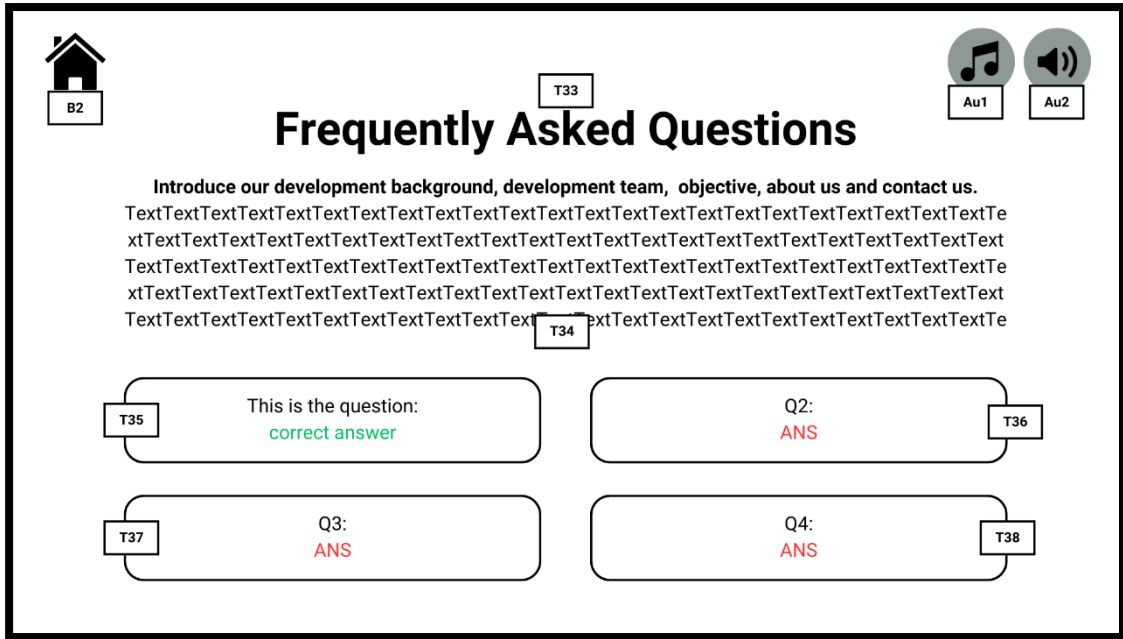
Figure 3.19 FAQ storyboard

Table 3.20 FAQ description and flow diagram

Description	Flow Diagram
B2: show the home button navigate to main menu page. **Au1:** show the music icon to turn on/off background music. **Au2:** show the audio icon to turn on/off the audio sound. **T33:** show the module function title. **T34:** show the text for introduce our team and objective. **T35:** show the text for the first FAQ. **T36:** show the text for the second FAQ. **T37:** show the text for the third FAQ. **T38:** show the text for the fourth FAQ.	

3.5 Project Planning

3.5.1 FYP project 1

3.5.1.1 Work Breakdown Structures (WBS)

Table 3.21 Project 1 WBS table

Level	WBS Name
1	**Introduction**
1.1	Background Study
1.2	Problems Statement
1.2.1	Identify problem statement
1.2.2	Refine problem statement
1.3	Set Project Objective
1.4	Project Scope
1.4.1	Define project scope
1.4.2	Refine project scope
1.5	Target User Identification
2	**Literature Review**
2.1	Similar application
2.1.1	Searching 5 similar application
2.1.2	Identify strengths and weakness
2.1.3	Compare the similar application
2.2	Analyze why Wetland Ecosystem important
2.3	Analyze Constructivism Approach difference
2.4	Analyze different between VLEs and convention learning
2.5	Users' requirements gathering and analysis
3	**Methodology**
3.1	Determine project methodology
3.1.1	Research software development methodology
3.1.2	Define ADDIE model

3.2	Software & Hardware requirement
3.3	Determine development platforms and tools
4	**Design phase**
4.1	System Flow Diagram
4.1.1	Design system flow diagram
4.1.2	Refine system flow diagram
4.2	Storyboard
4.2.1	Design storyboard
4.2.2	Refine storyboard
4.3	Prototype
4.3.1	Design prototype
4.3.2	Refine prototype

3.5.1.2 Gantt Chart
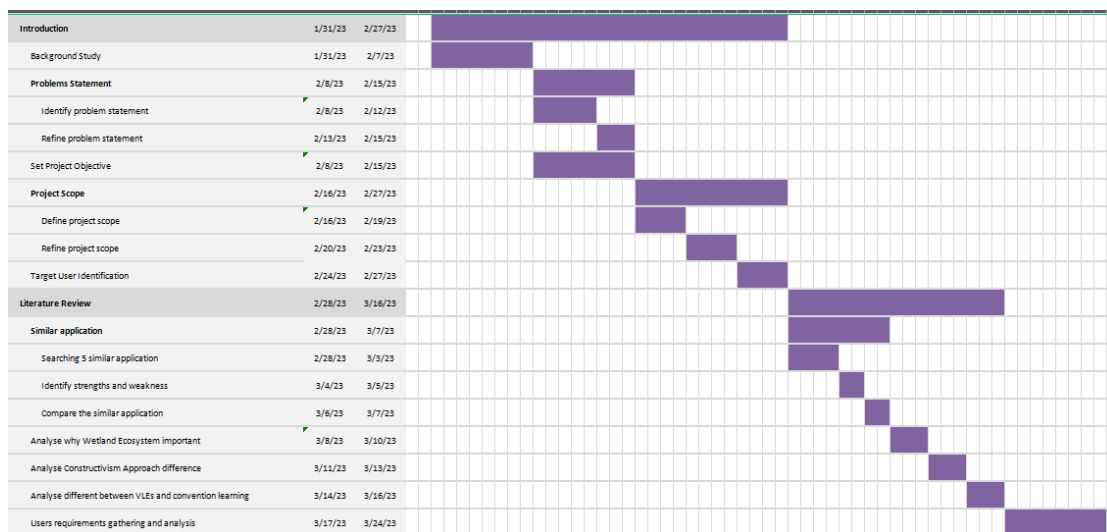
Figure 3.20 Gantt Chart 1

Figure 3.21 Gantt Chart 2

3.5.2 FYP project 2

3.5.2.1 Work Breakdown Structures (WBS)

Table 3.22 Project 2 WBS table

Level	WBS Name
5	**Development phase**
5.1	Develop 5 module
5.1.1	Simulation module
5.1.2	Theory-base module
5.1.3	Mini game module
5.1.4	Assessment module
5.1.5	Feedback module
5.2	Authoring
5.2.1	System compilation and integration
5.2.2	Check and finalize
6	**Testing**
6.1	Unit Testing
6.2	Integration Testing
6.3	System Testing
6.4	User Acceptance Testing

3.5.2.2 Gantt Chart
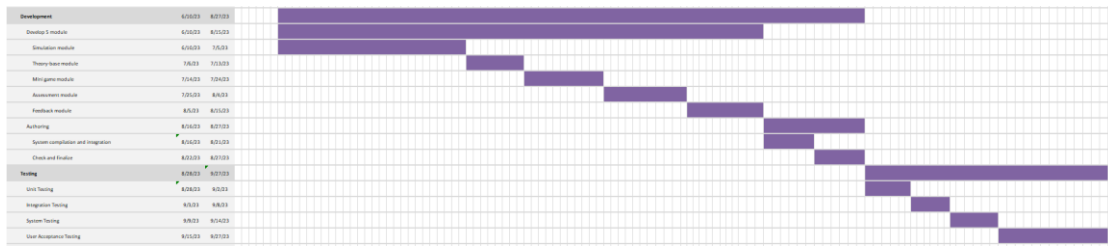
Figure 3.22 Gantt chart 3

CHAPTER 4

DEVELOPMENT

4.1 Overview

This software development project represents a comprehensive and multifaceted endeavor that leverages a range of tools and technologies to achieve its objectives. Unity serves as the central platform for the project, encompassing various facets such as user interface (UI) development, the creation of immersive 3D environments, and the integration of diverse project components. This software development initiative combines the capabilities of Unity, Visual Studio, Blender, phpMyAdmin, and Figma to create an innovative and immersive learning application for wetland ecology. Through its constructivist approach and rigorous evaluation, the project aims to enhance the accessibility and effectiveness of wetland ecology education for a diverse audience.
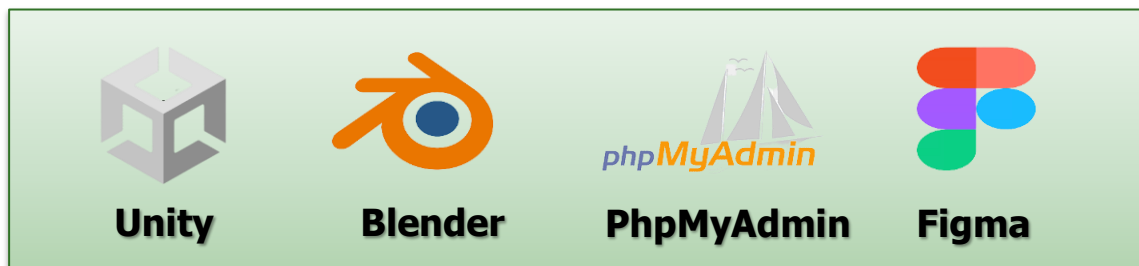
Figure 4.1 Software Development Tools

4.2 Development Process

4.2.1 Maine Menu

Main Menu
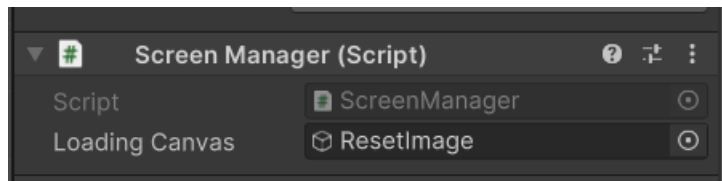
Figure 4.2 Scree Manager in Unity

```csharp
using UnityEngine;
using UnityEngine.SceneManagement;

public class ScreenManager : MonoBehaviour
{

    [SerializeField] private GameObject loadingCanvas;

    public void startGame()
    {
        SceneManager.LoadScene(1);
    }

    public void TheoryModule()
    {
        SceneManager.LoadScene(2);
    }
}
```

Figure 4.3 Screen Manager Script

The ScreenManager script facilitates user navigation between various pages within a Unity application. To get started, you should first import the Unity Engine Scene Management system, which grants you access to functions like LoadScene for transitioning between scenes. Next, you can create custom functions within your script and use SceneManager.LoadScene(id) to navigate to different pages or scenes. When you mark a variable with [SerializeField], you're making that variable visible and editable in the Unity Inspector even if it's private.
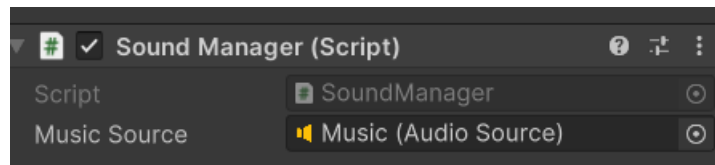
Figure 4.4 Sound Manager in Unity

```csharp
public class SoundManager : MonoBehaviour
{
    public static SoundManager Instance;

    [SerializeField] private AudioSource _musicSource;

    private bool isPlayingMusic = false; // Add a flag to track if music is currently playing

    private void Awake()
    {
        if (Instance == null)
        {
            Instance = this;
            DontDestroyOnLoad(gameObject);
        }
        else
        {
            Destroy(gameObject);
        }
    }

    private void Start()
    {
        PlayMusic();
    }

    public void PlayMusic()
    {
        _musicSource.Play();
        isPlayingMusic = true; // Set the flag when music starts playing
    }

    // Add this method to stop music
    public void StopMusic()
    {
        _musicSource.Stop();
        isPlayingMusic = false; // Set the flag when music stops
    }
}
```

Figure 4.5 Sound Manager Script

The SoundManager script is designed to provide a consistent background music experience across all screens in your application. By utilizing the SerializeField attribute, you can easily assign Unity game objects through a simple drag-and-drop process. In the SoundManager, the Unity game object's AudioSource component plays a pivotal role in activating the desired sound.

Within the script's Start function, the PlayMusic function is invoked, responsible for initiating playback of the AudioSource with specified parameters. Conversely, the StopMusic function, which operates similarly to PlayMusic, is responsible for halting the audio playback. Furthermore, the ChangeMasterVolume function is used to dynamically adjust the AudioSource's volume based on the input from a slider component. The Toggle Music function allows users to switch the music on or off with a simple click of the toggle music button. Meanwhile, the "Music Volume" function empowers users to modify the audio source's volume by interacting with a slider component.
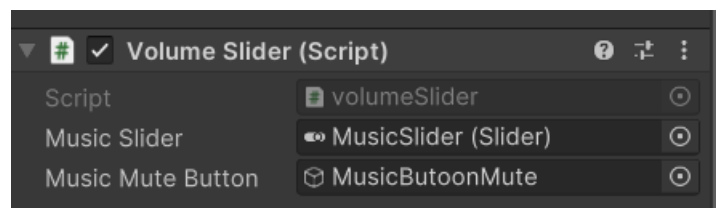
Figure 4.6 Volume Slider in Unity

```csharp
public class volumeSlider : MonoBehaviour
{
    [SerializeField] public Slider musicSlider;
    [SerializeField] private GameObject musicMuteButton; // Reference to the mute button for music

    void Start()
    {
        // Get the initial music volume from the SoundManager
        float initialMusicVolume = SoundManager.Instance.GetMusicVolume();

        // Set the initial slider value to match the initial music volume
        musicSlider.value = initialMusicVolume;

        // Set up the listener to update the volume when the slider changes
        musicSlider.onValueChanged.AddListener(val =>
        {
            SoundManager.Instance.MusicVolume(val);
            Debug.Log($"New volume value: {val}");
        });
    }
}
```

Figure 4.7 Volume Slider Script

The volumeSlider script empowers users to establish a music slider for regulating audio volume. By using the SerializeField attribute, developers can conveniently assign the slider component to the script through drag-and-drop functionality.

Within this script, the Start function plays a pivotal role in obtaining the current music volume. It achieves this by invoking the GetMusicVolume function from the SoundManager. Subsequently, the obtained volume value is applied to the slider,

synchronizing it with the audio source's current volume setting. The ToggleMusic function is responsible for triggering the ToggleMusic function within the SoundManager script. Additionally, it toggles the appearance of the button, transforming it from a Toggle Music button to a "Mute Music" button.
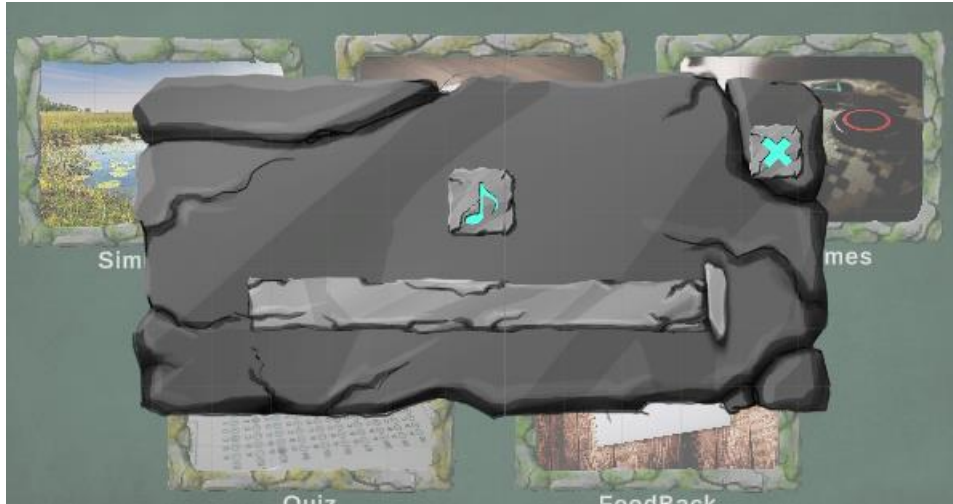
Figure 4.8 Volume Slider Screen Shot
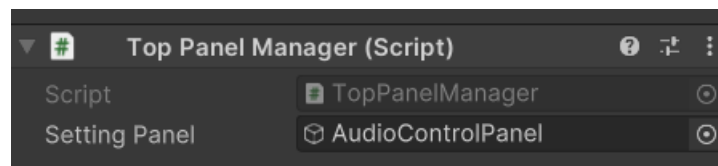
Figure 4.9 Top Panel Manager in Unity

```csharp
public class TopPanelManager : MonoBehaviour
{
    [SerializeField] private GameObject SettingPanel;

    // Start is called before the first frame update
    public void ShowSettingPanel()
    {
        SettingPanel.SetActive(true);
    }

    public void CloseSettingPanel()
    {
        SettingPanel.SetActive(false);
    }
}
```

Figure 4.10 Top Panel Manager Script

Within the TopPanelManager script, the management of the settings button is handled. This script comprises two distinct functions: ShowSettingPanel and CloseSettingPanel. The ShowSettingPanel function is responsible for displaying the

settings panel when triggered. This serves as the first function. Inside the settings panel, the CloseSettingPanel function takes charge of closing the settings panel. This serves as the second function.

Figure 4.11 Main Menu Screen Shot

4.2.2 Module 1

Simulation module

Within this project, the simulation module encompasses a diverse ecosystem featuring three distinct animals: crocodile, heron, and duck, along with two types of plants: water lily and cattail. Each animal is equipped with three essential functions: eat, mate, and sound. The plant life, on the other hand, comprises two significant functions: pollution and growth.

In addition to the animals and plants, there is a Count Panel incorporated into the simulation. This panel serves the purpose of displaying the current population count of the various species. To enhance user engagement and interaction with the simulation, three buttons have been provided:

- A "Switch Day and Night" button, allowing users to toggle between daytime and nighttime conditions.
- A "Raining" button, enabling users to introduce rain into the simulation environment.

- A "Reset" button, permitting users to reset the simulation, thereby facilitating exploration and experimentation within the ecosystem.
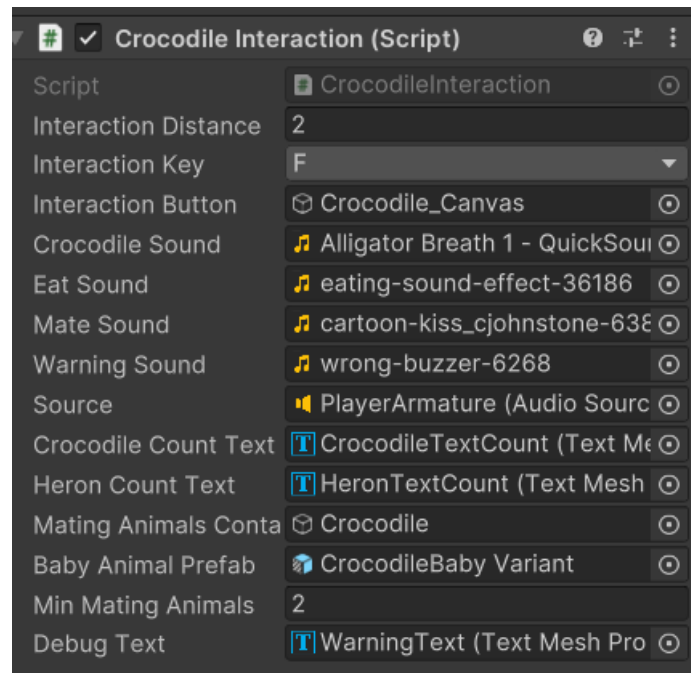
Figure 4.12 Crocodile Interaction in Unity

The CrocodileInteraction script is responsible for facilitating various interactions with crocodiles, including their eating, mating, producing sounds, and population count functions. These interactions allow users to engage with the crocodile species and observe their responses.

The script begins with the Start function, which initiates the crocodile count and identifies the number of crocodiles that feed on herons. This count is then displayed in the top right-hand corner of the screen. Subsequently, the Update function continuously monitors the proximity of animals to the user character. When an animal comes close, an interaction panel becomes visible. By pressing the appropriate key, users can engage in different interactions with the crocodiles, including feeding, mating, or eliciting sounds from them.

```
void Start()
{
    CountCrocodileAnimals();
    CountfeedAnimals();
}


void Update()
{
    // Check if the player is near an animal
    CheckNearAnimal();

    // Display or hide the key prompt based on distance
    interactionButton.gameObject.SetActive(isNearAnimal);

    // Update the key prompt text
    if (isNearAnimal)
    {
        interactionButton.gameObject.SetActive(true);
    }
    // Handle interaction button click
    if (isNearAnimal && Input.GetKeyDown(KeyCode.L))
    {
        InteractWithAnimalSound();
    }
    if (isNearAnimal && Input.GetKeyDown(KeyCode.E))
    {
        Eat();
    }

    if (isNearAnimal && Input.GetKeyDown(KeyCode.M))
    {
        Mate();
    }

}
```

Figure 4.13 Animal Interaction Script Start and Update

Both the CountCrocodileAnimals and CountFeedAnimals functions serve the purpose of tallying the number of objects based on their assigned tags. Once the counting process is completed, the numerical count is then promptly updated and displayed on the top right-hand corner of the screen. These counting functions are integrated into both the "mate" and "eat" functions to ensure a consistent update of counts as interactions occur.

```csharp
private int CountCrocodileAnimals()
{
    int crocodileCount = 0;
    //int crocodileCountText = 0;

    // Find all crocodile animals in the scene
    GameObject[] crocodileAnimals = GameObject.FindGameObjectsWithTag("Crocodile");

    // Count the number of crocodile animals
    crocodileCount = crocodileAnimals.Length;
    //crocodileCountText = crocodileAnimals.Length+1;

    CrocodileCountText.text = "Crocodile: " + crocodileCount;

    return crocodileCount;
}

private int CountfeedAnimals()
{
    int feedCount = 0;

    // Find all crocodile animals in the scene
    GameObject[] feedAnimals = GameObject.FindGameObjectsWithTag("Heron");

    // Count the number of crocodile animals
    feedCount = feedAnimals.Length;

    HeronCountText.text = "Heron: " + feedCount;

    return feedCount;
}
```

Figure 4.14 Animal Interaction Script Count Function

The "DebugMessage" function is designed to exhibit messages prominently at the center of the screen. This function is activated in specific scenarios, such as when the "eat" function encounters a situation where no more animals can be consumed.

To maintain a clutter-free display, the "ClearDebugText" function has been implemented. This function serves to remove the text message from the screen after a brief duration, ensuring that the display remains unobtrusive.

```csharp
private void DebugMessage(string message)
{
    // Display the debug message on the UI Text element
    debugText.text = message;
}

private IEnumerator ClearDebugText(float delay)
{
    yield return new WaitForSeconds(delay);
    debugText.text = string.Empty; // Clear the debug text
}
```

Figure 4.15 Animal Interaction Message Function
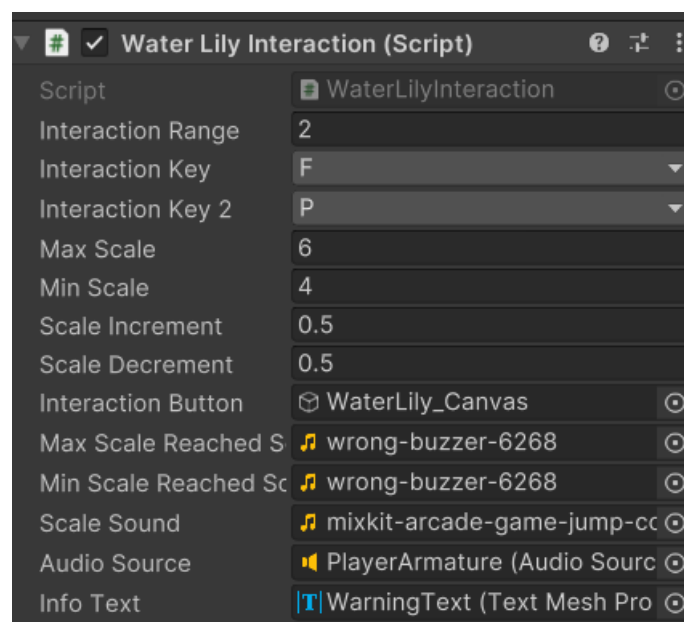
Figure 4.16 Water Lily Interaction in Unity

The Water Lily Script comprises two primary functions: ScaleWaterLilyInScene and PollutionWaterLiliesInScene. These functions require the user to approach the plant closely, triggering the appearance of an interaction panel. Upon activation, the user can utilize the interaction key to either promote growth or address pollution concerns.

The first function, ScaleWaterLilyInScene, facilitates the enlargement of the plant based on its object tag. Each plant possesses a predefined maximum scale limit, and if the plant reaches this limit, an error message is displayed. Conversely, the second function, PollutionWaterLiliesInScene, reduces the plant's scale according to its

object tag. Each plant also has a minimum scale limit, and when the plant reaches this threshold, an error message is generated to notify the user.

```
void Update()
{
    // Check if the player is near an animal
    CheckNearPlant();

    // Display or hide the key prompt based on distance
    interactionButton.gameObject.SetActive(isNearPlant);

    // Update the key prompt text
    if (isNearPlant)
    {
        interactionButton.gameObject.SetActive(true);
    }

    // Handle interaction button click
    if (isNearPlant && Input.GetKeyDown(interactionKey))
    {
        Debug.Log("Grow Up");
        ScaleWaterLilyInScene();
    }

    if (isNearPlant && Input.GetKeyDown(interactionKey2))
    {
        Debug.Log("Pollution Plant");
        //ScaleWaterLilyInScene2();
        PollutionWaterLiliesInScene();
    }
}
```

Figure 4.17 Water Lily Interaction Script

Lighting Toggle (Script)		
Script	LightingToggle	
Sun Light	MorningSun (Light)	
Button Text TMP	Text (TMP) (Text Mesh Pro U	
Interaction Script 1	PlayerArmature (Crocodile In	
Interaction Script 2	PlayerArmature (Heron Intera	
Interaction Script 3	PlayerArmature (Duck Interac	
Interaction Script 4	PlayerArmature (Cattails Inte	
Interaction Script 5	PlayerArmature (Water Lily In	

Figure 4.18 Lighting Toggle in Unity

The primary function of the Lighting Toggle Script, known as ToggleLighting, enables users to adjust the light intensity through interaction keys at any location within the application. When the light intensity is increased, indicating daytime, users can interact with animals, and animal animations become active. Conversely, when

the light intensity is decreased, symbolizing nighttime, the animal and plant interaction scripts become inactive, and animal animations cease to function.

```
public void ToggleLighting()
{
    isDay = !isDay;
    enableAnimations = !enableAnimations;

    if (isDay)
    {
        // Set the lighting to morning (increase intensity, change color, etc.)
        sunLight.intensity = 3.0f;
        buttonTextTMP.text = "Morning"; // Update the TMP text
        interactionScript1.enabled = true;
        interactionScript2.enabled = true;
        interactionScript3.enabled = true;
        interactionScript4.enabled = true;
        interactionScript5.enabled = true;
    }
    else
    {
        // Set the lighting to night (decrease intensity, change color, etc.)
        sunLight.intensity = 1.0f;
        buttonTextTMP.text = "Night"; // Update the TMP text
        interactionScript1.enabled = false;
        interactionScript2.enabled = false;
        interactionScript3.enabled = false;
        interactionScript4.enabled = false;
        interactionScript5.enabled = false;
    }

    AnimalAnimatorControl(enableAnimations);
}
```

Figure 4.19 Lighting Toggle Script
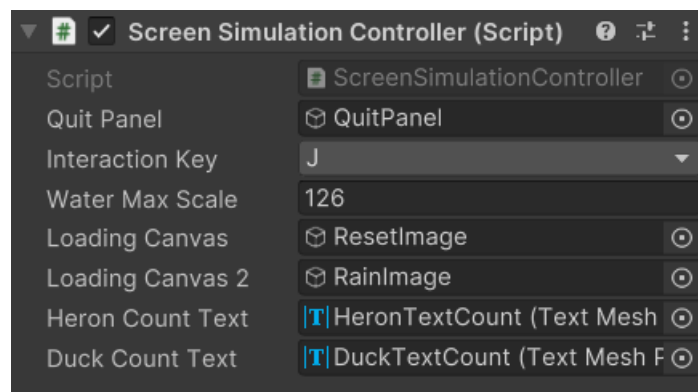
Figure 4.20 Screen Simulation Controller in Unity

The Screen Simulation Controller Script encompasses three primary functions: BackToMenu, ResetSimulation, and ScaleWaterInScene. The initial function, BackToMenu, becomes active upon the user's interaction keypress "H," prompting the display of a quit panel. Within this panel, the user must confirm their intention to quit

by pressing "Y" to return to the main menu or "N" to dismiss the panel and resume the simulation module.

The second function, ResetSimulation, comes into play when the user presses the "R" key. This function resets all components to their initial state, effectively restarting the simulation. Lastly, the ScaleWaterInScene function allows the user to adjust the water level by pressing the "J" key. These three functions create interactive opportunities for users to observe and explore the dynamics and relationships within the simulation module.

```csharp
void Update()
{
    // Handle interaction button click
    if (Input.GetKeyDown(KeyCode.H))
    {
        QuitPanel.SetActive(true);
        isQuitting = true;
    }

    if (isQuitting)
    {
        if (Input.GetKeyDown(KeyCode.Y))
        {
            BackToMenu(); // Quit the application
        }
        else if (Input.GetKeyDown(KeyCode.N))
        {
            // Hide the quit panel and reset the quitting flag
            QuitPanel.SetActive(false);
            isQuitting = false;
        }
    }

    if (Input.GetKeyDown(KeyCode.R))
    {
        ResetSimulation();
    }

    if (enableMouseControl)
    {
        HandleMouseControl();
    }
    else
    {
        HandleKeyboardControl();
    }

    if (Input.GetKeyDown(interactionKey))
    {
        Debug.Log("Water up");
        ScaleWaterInScene(3.0f);
    }
}
```

Figure 4.21 Screen Simulation Script
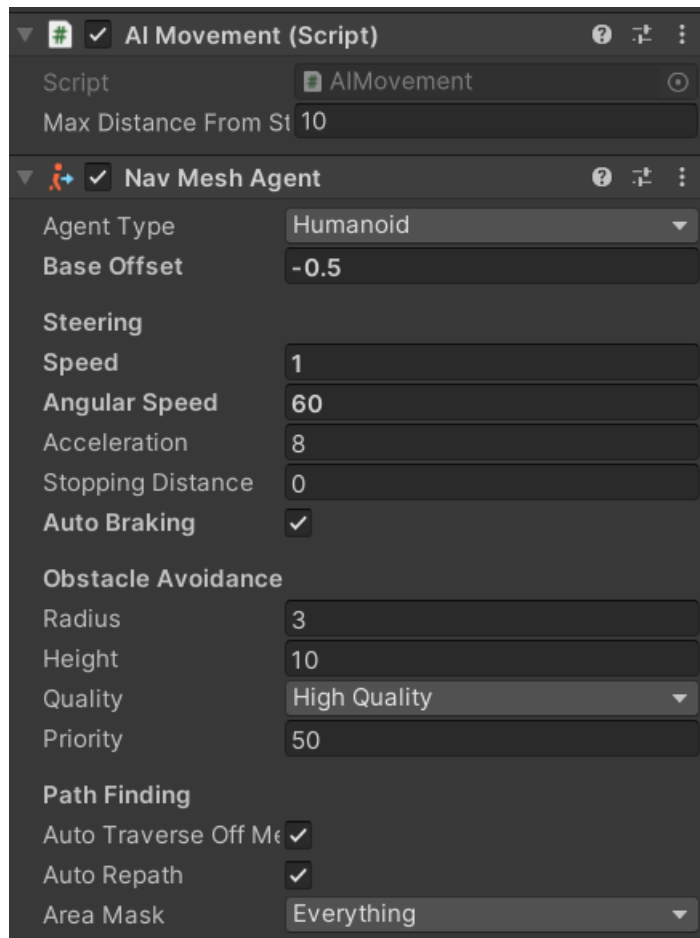
Figure 4.22 AI Movement in Unity

This AI movement script enables animals to autonomously navigate within a confined area, leveraging the capabilities of Unity's AI support and Nav Mesh Agent component. This approach streamlines the movement control process and enhances the realism of in-game animal behavior.

```csharp
using UnityEngine.AI;

public class AIMovement : MonoBehaviour
{

    private bool isDay = true; // Variable to track day/night status

    private NavMeshAgent agent;
    private Vector3 randomTarget;
    private Vector3 startingPosition;
    public float maxDistanceFromStartingPosition = 10f; // Maximum distance from the starting position


    private void Start()
    {
        agent = GetComponent<NavMeshAgent>();
        startingPosition = transform.position; // Store the starting position
        SetRandomTargetWithinBounds();
    }
```

Figure 4.23 AI Movement Script
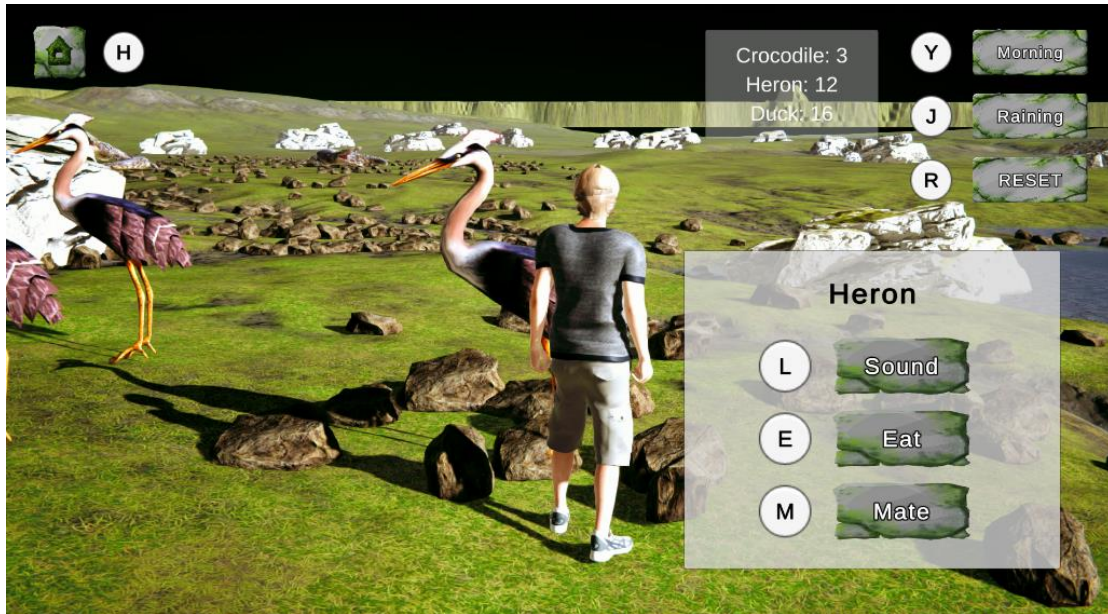
Figure 4.24 Simulation Module Screen Shot

4.2.3 Module 2

Theory module

- Important of wetland (using dialog function to show the theory information)
- Wetland map (interact with the animal image and each animal have 3 ecosystem information buttons provide to user)

Important of wetland
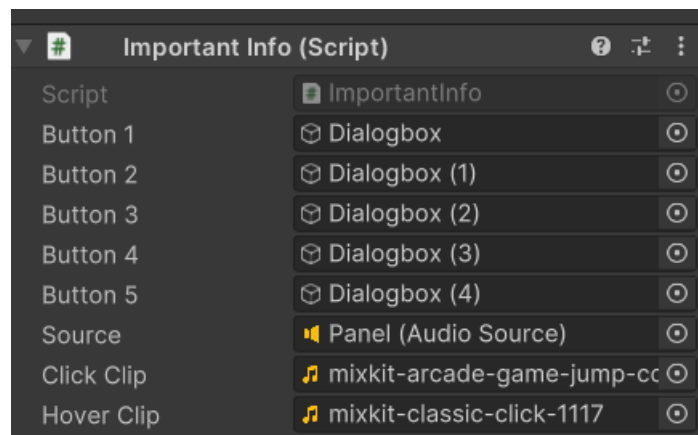
Figure 4.25 Important Info in Unity

In this particular sub-module focusing on the significance of wetlands, the central feature is a dialog box. This dialog box is constructed using essential Unity UI

components, consisting of an image serving as the dialog box's background and text providing the dialog's content. Within the script, the "showInfo()" function plays a pivotal role by activating the dialog box. In the user interface, five buttons are prominently displayed, each imparting essential information about wetlands. Users have the opportunity to interact with these five buttons to access specific knowledge regarding wetlands.

These five buttons have been thoughtfully configured with the "onClick()" attribute, directing them to respond to specific scripts and functions. When a user clicks on one of these buttons, the dialog box becomes visible, allowing users to access and absorb the pertinent information related to wetlands.

```
public void showInfo1()
{

    Button1.SetActive(true);
    //LeanTween.moveLocal(Button1, new Vector3(0, 0, 0), 0.5f).setDelay(0.1f).setOnComplete(onCompleteCallback.Invoke);

}

public void showInfo2()
{

    Button2.SetActive(true);
    //LeanTween.moveLocal(Button2, new Vector3(0, 0, 0), 0.5f).setDelay(0.1f).setOnComplete(OnComplete);

}
```

Figure 4.26 Important Info Script
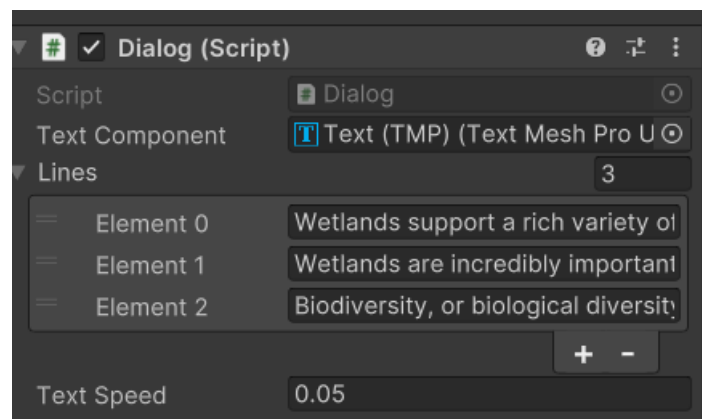
Figure 4.27 Dialog in Unity

Within the "DialogScript," you have the flexibility to generate multiple dialog elements, and developers can populate each of these elements with their desired text in the Unity Inspector. Upon starting the script, the text content is presented in a sequential manner, displaying one line at a time within the first element. As soon as

the text in the first element is fully displayed, it automatically transitions to the next element, where the text is again shown line by line.

During the presentation of text within an element, if the text is still being revealed but not yet complete, a simple mouse click triggers the immediate display of the entire content for that element. If there are additional elements, the script seamlessly continues to present the text in the next element, following the same line-by-line approach until all elements have been presented.

```csharp
public class Dialog : MonoBehaviour
{
    public TextMeshProUGUI textComponent;
    public string[] lines;
    public float textSpeed;

    private int index = 0;

    // Start is called before the first frame update
    void Start()
    {
        textComponent.text = string.Empty;
        StartDialogue();
    }

    // Update is called once per frame
    void Update()
    {
        if(Input.GetMouseButtonDown(0))
        {
            if(textComponent.text==lines[index])
            {
                nextLine();
            }
            else
            {
                StopAllCoroutines();
                textComponent.text = lines[index];
            }
        }
    }

    void StartDialogue()
    {
        index = 0;
        StartCoroutine(TypeLine());
    }
```

Figure 4.28 Dialog Script
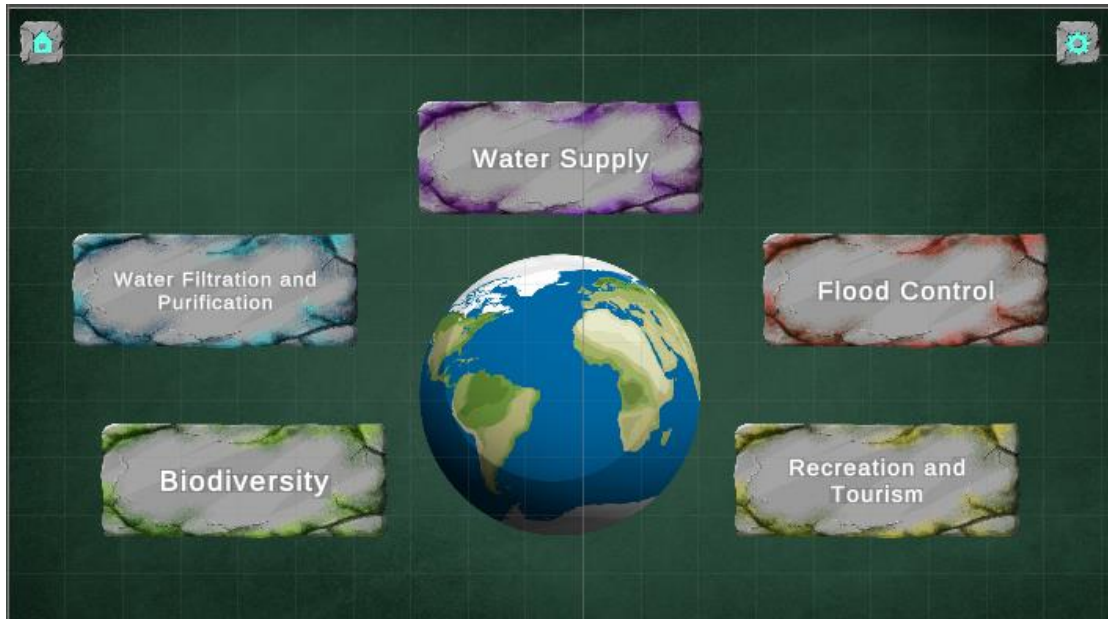
Figure 4.29 Important of Wetland Screen Shot

Theory Map
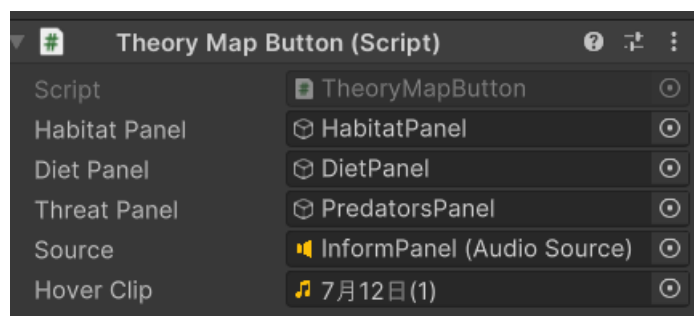
Figure 4.30 Theory Map Button in Unity

The TheoryInformationButton script similar to screen manager using Load.Secene()
and the sound source.PlayOneShot function. Within the "TheoryMapButton" script,
the focus is on conveying three key categories of information: habitat, diet, and
predators. For each of these categories, two distinct functions have been established to
facilitate the display and closure of the information panel. Additionally, the script
incorporates an AudioSource and AudioClip, serving as components for audio
playback. A specific function has been implemented to trigger the sound's playback.

In our application, when users interact with this button, we provide them with
information related to habitat, diet, and predators. Furthermore, we have integrated an
auditory component to enhance the user experience. Specifically, when users hover

their mouse over the button, a hover sound is played, and when they move the mouse away, the information panel is closed.

```csharp
public class TheoryMapButton : MonoBehaviour
{
    [SerializeField] private GameObject habitatPanel, dietPanel, threatPanel;

    [SerializeField] private AudioSource source;
    [SerializeField] private AudioClip hoverClip;

    private GameObject HabitatPanel { get => habitatPanel; }
    private GameObject DietPanel { get => dietPanel; }
    private GameObject ThreatPanel { get => threatPanel; }
    // Start is called before the first frame update

    public void showHabitatInfo() {

        HabitatPanel.SetActive(true);

    }

    public void showDietInfo()
    {

        DietPanel.SetActive(true);

    }

    public void showThreatInfo()
    {

        ThreatPanel.SetActive(true);

    }

    public void closeHabitatInfo()
    {

        HabitatPanel.SetActive(false);

    }
```

Figure 4.31 Theory Map Button Script
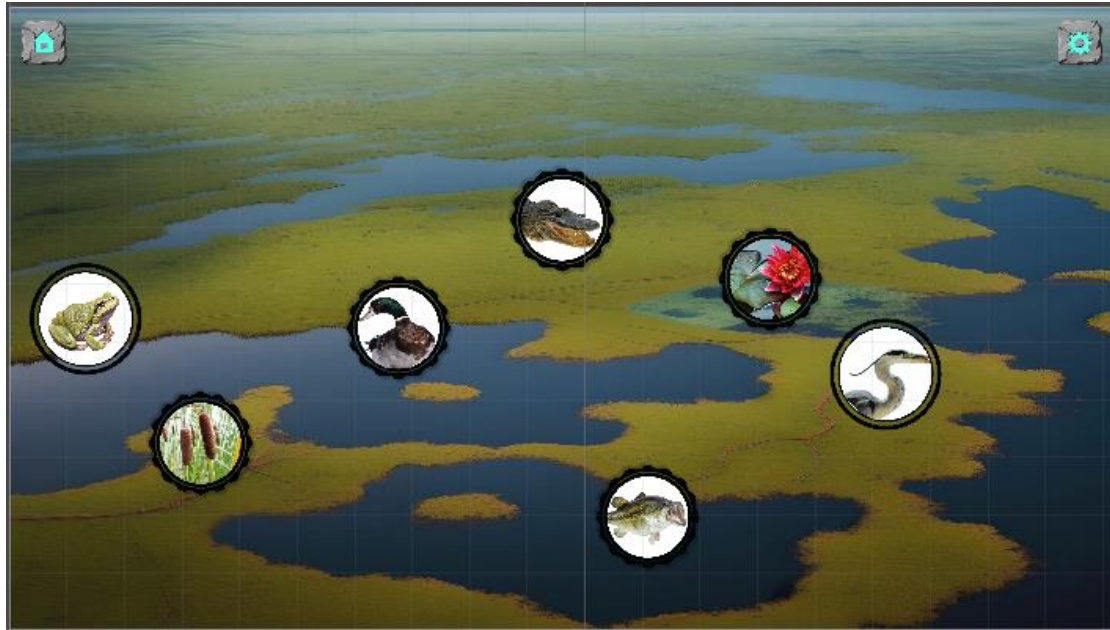
Figure 4.32 Theory Map Screen Shot

4.2.4 Module 3

Mini game module

Matching Card Game

- In this game, users have the ability to select two cards consecutively. If the first card matches with the second card, those cards are successfully paired and removed from the game. However, if the matching attempt fails, the cards are turned back to their initial positions. A side panel keeps track of the user's click count and the time elapsed, encouraging users to improve their performance.

Drag and Drop Game

- In this game, users have the ability to select two cards consecutively. If the first card matches with the second card, those cards are successfully paired and removed from the game. However, if the matching attempt fails, the cards are turned back to their initial positions. A side panel keeps track of the user's click count and the time elapsed, encouraging users to improve their performance.

MATCHING GAME
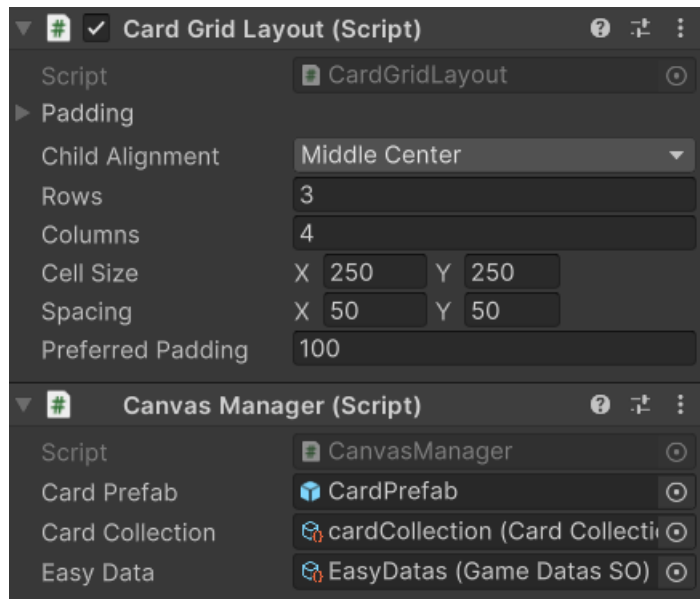
Figure 4.33 Card Grid Layout and Canvas Manager in Unity

In the CardGridLayout script, its primary purpose is to dynamically generate the layout for the card grid. The CalculateLayoutInputHorizontal function is responsible for determining the optimal arrangement, which includes calculating the number of rows and columns, determining cell sizes, specifying spacing, and applying padding to create an organized card layout.


```
public class CardGridLayout : LayoutGroup
{
    public int rows;
    public int columns;
    public Vector2 cellSize;
    public Vector2 spacing;
    public int preferredPadding;

    public override void CalculateLayoutInputHorizontal()
    {
        base.CalculateLayoutInputHorizontal();
        if (rows == 0 || columns == 0)
        {
            rows = 4;
            columns = 5;
        }
```

Figure 4.34 Card Grid Layout Script

The CanvasManager script is dedicated to the card grid generation process. It configures the grid layout and oversees card generation. This script effectively gathers card data and prefabs to facilitate the card generation system.

```
public class CanvasManager : MonoBehaviour
{
    public GameObject cardPrefab;
    public CardCollectionSO cardCollection;

    public GameDatasSO easyData;
    //public GameDatasSO normalData;
    //public GameDatasSO hardData;

    GameDatasSO gameDatas;
    CardGridGenerator cardGridGenerator;

    List<CardController> cardControllers;

    void Awake()
    {
        cardControllers = new List<CardController>();
        GetGameDatasByDifficulty();

        cardGridGenerator = new CardGridGenerator(cardCollection, gameDatas);

        SetCardGridLayoutParams();
        GenerateCards();

        GameManager gameManager = (GameManager)FindObjectOfType(typeof(GameManager));
        gameManager.CardCount = gameDatas.rows * gameDatas.columns;
    }
}
```

Figure 4.35 Canvas Manager Script
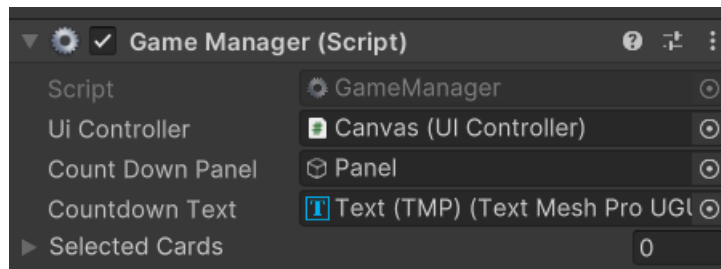
Figure 4.36 Game Manager in Unity

Within the "GameManager" script, the central function is "InitStates," which sets up all the required states for the matching game to run smoothly. These states play a crucial role in managing the game's flow and interactions. At the outset, the countdown panel becomes active to introduce a cutscene and prepare the user for the upcoming gameplay experience.

```
void InitStates()
{
    cardSelectionState = new CardSelectionState(this);
    pairSelectionState = new PairSelectionState(this);
    memorizeCardsState = new MemorizeCardsState(this, 0.5f);
    matchingCardsState = new MatchingCardsState(this, 0.2f);
    pauseGameState = new PauseGameState(this);
    endGameState = new EndGameState(this);

    gameState = cardSelectionState;
}

public void TransitionState(GameState newState)
{
    gameState.EndState();
    gameState = newState;
    gameState.EnterState();
}
```

Figure 4.37 Game Manager Script
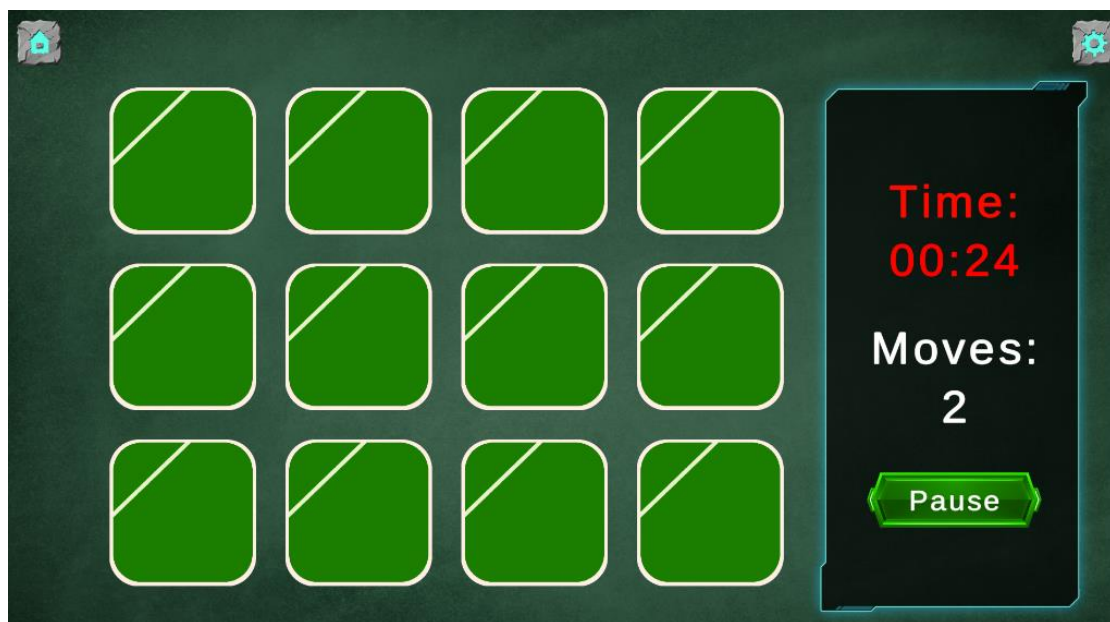
Figure 4.38 Matching Game Screen Shot
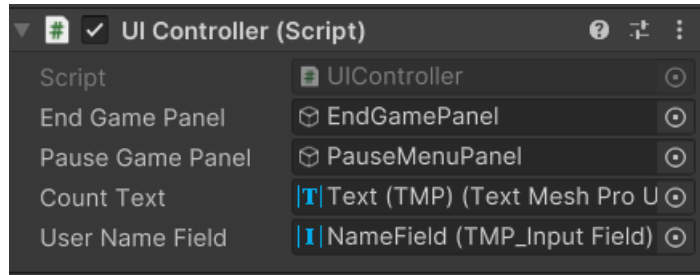
Figure 4.39 UI Controller in Unity

Within the "UIController" script, the primary responsibility lies in managing the end game and pause game panels. When users click the pause button, the "ActivePausePanel" function is triggered, activating the pause panel and enabling game pausing. Upon game completion, invoking the "ActiveEndPanel" function activates the end panel, providing closure to the game and displaying relevant information. Additionally, the script includes a "QuitToMenu" function, which allows users to return to the previous scene or page by loading the scene corresponding to Unity's build index.

```
public class UIController : MonoBehaviour
{
    public GameObject endGamePanel;
    public GameObject pauseGamePanel;


    public TMP_Text countText;

    public TMP_InputField userNameField;

    int movesCount;

    // Start is called before the first frame update
    void Start()
    {

        endGamePanel.SetActive(false);
        pauseGamePanel.SetActive(false);

        countText.text = "0";
    }
```

Figure 4.40 UI Controller Script
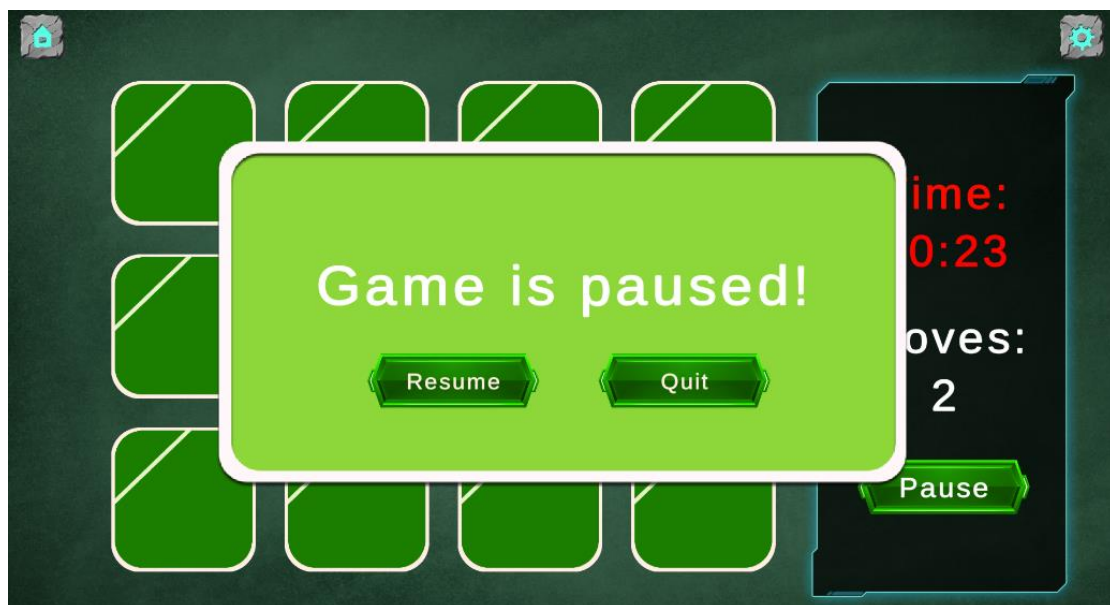
Figure 4.41 Pause Panel Screen Shot

DRAG AND DROP
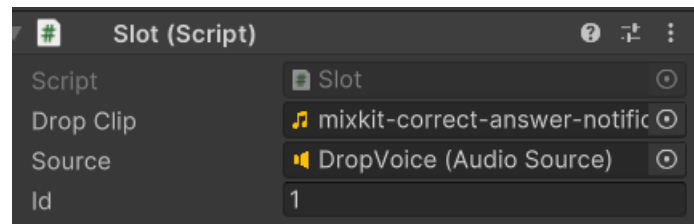
Figure 4.42 Slot in Unity

The Slot script is configured with three essential parameters: dropClip, source, and id. The first two parameters, dropClip and source, are primarily employed to generate auditory feedback in response to mouse clicks.

The third parameter, id, plays a crucial role in determining the matching logic. When the "id" matches the "id" value from the DragAndDrop script, the script takes specific actions. This includes anchoring the position, playing a designated drop sound, and incrementing the points on the WinPoints game object by one. In case of a mismatch, the script resets the position to its original state.

```csharp
public class Slot : MonoBehaviour, IDropHandler
{
    // Start is called before the first frame update
    [SerializeField] private AudioClip dropClip;
    [SerializeField] private AudioSource source;
    public int id;

    public void OnDrop(PointerEventData eventData)
    {
        Debug.Log("Item Dropped");

        if (eventData.pointerDrag != null)
        {
            if(eventData.pointerDrag.GetComponent<DragAndDop>().id == id)
            {
                eventData.pointerDrag.GetComponent<RectTransform>().anchoredPosition = this.GetComponent<RectTransform>().anchoredPosition;
                source.PlayOneShot(dropClip);
                GameObject.Find("WinPoints").GetComponent<WinPage>().AddPoints();
            }
            else
            {
                eventData.pointerDrag.GetComponent<DragAndDop>().ResetPosition();
            }

        }
    }
}
```

Figure 4.43 Slot Script
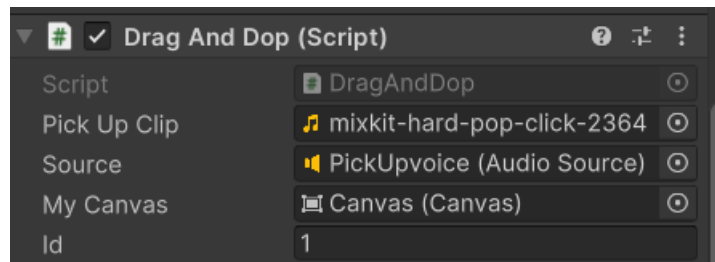
Figure 4.44 Drag and Drop in Unity

The DragAndDrop script implements essential mouse drag and drop functionality using PointerEventData. These functions, namely OnBeginDrag, OnDrag, OnEndDrag, and OnPointerDown, empower users to interact with game objects through dragging and dropping actions. The ResetPosition function plays a vital role by restoring objects to their original positions in the event of an incorrect answer.

Within this script, four parameters are defined: pickUpClip, source, canvas, and id. The first two parameters, pickUpClip and source, contribute to creating auditory feedback upon mouse clicks. The canvas parameter is essential for preserving position data, ensuring that the reset function operates smoothly. Lastly, the id parameter serves as a matching identifier, allowing authors to assign unique IDs to image canvases and establish corresponding associations.

```
void Start()
{
    rectTrans = GetComponent<RectTransform>();
    canvasGroup = GetComponent<CanvasGroup>();
    initPos = transform.position;


}
public void OnBeginDrag(PointerEventData eventData)
{
    Debug.Log("BeginDrag");
    canvasGroup.blocksRaycasts = false;
}


public void OnDrag(PointerEventData eventData)
{
    //Debug.Log("OnDrag");
    rectTrans.anchoredPosition += eventData.delta / myCanvas.scaleFactor;
}


public void OnEndDrag(PointerEventData eventData)
{
    Debug.Log("EndDrag");
    canvasGroup.blocksRaycasts = true;
}


public void OnPointerDown(PointerEventData eventData)
{
    float volume = 0.2f;
    source.PlayOneShot(pickUpClip, volume);
}


public void ResetPosition()
{
    transform.position = initPos;
}
```

Figure 4.45 Drag and Drop Script
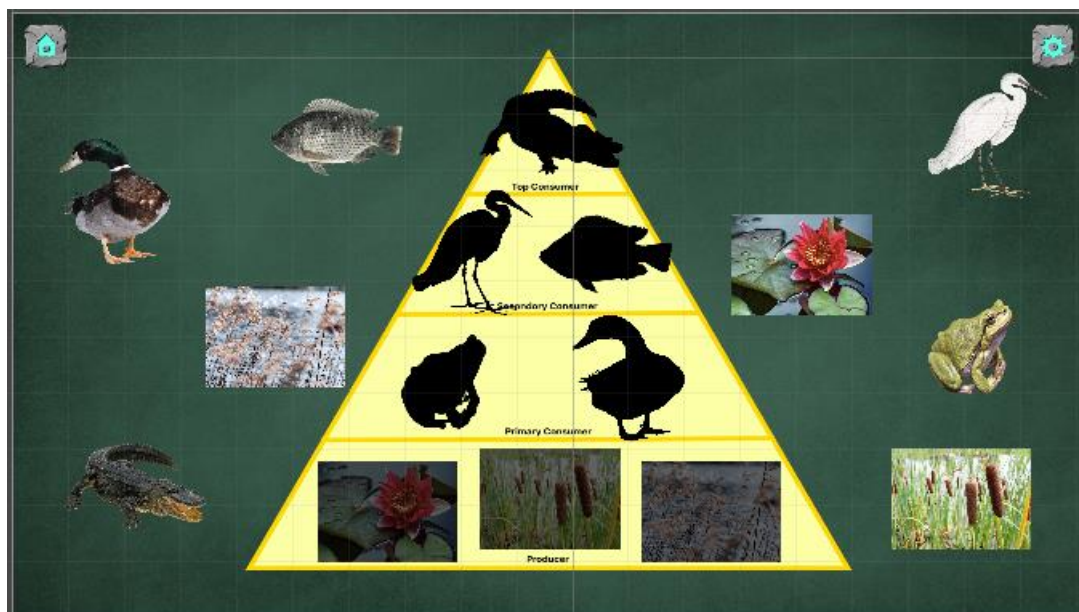
Figure 4.46 Drag and Drop Screen Shot
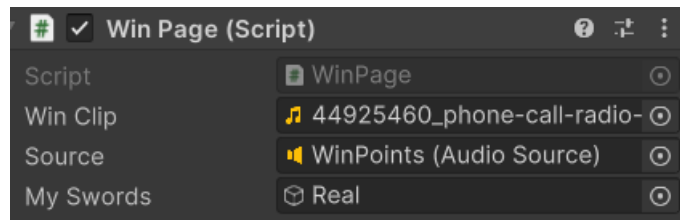
Figure 4.47 Win Page in Unity

In the WinPage script, the central function is AddPoints, which serves the primary purpose of revealing the win panel. When the script is initiated, the "Start" function calculates the number of child elements within the game object and stores this count in the pointsToWin parameter.

As the AddPoints function is executed, it continuously increments the current points by one with each activation. When the current points match the value stored in pointsToWin, a designated win sound, played via the source component, is triggered. Simultaneously, the win panel becomes visible, signaling successful completion of the game.

```
public class WinPage : MonoBehaviour
{
    [SerializeField] private AudioClip WinClip;
    [SerializeField] private AudioSource source;
    private int pointsToWin;
    private int currentPoints;
    public GameObject mySwords;
    // Start is called before the first frame update
    void Start()
    {
        pointsToWin = mySwords.transform.childCount;
    }

    // Update is called once per frame
    void Update()
    {

    }

    public void AddPoints()
    {
        currentPoints++;

        if (currentPoints >= pointsToWin)
        {
            source.PlayOneShot(WinClip);
            transform.GetChild(0).gameObject.SetActive(true);
        }
    }
}
```

Figure 4.48 Win Page Script

Figure 4.49 Win Panel Screen Shot

4.2.5 Module 4

Quiz module

In the Quiz module, users are presented with questions that encompass various formats, including text, audio, images, and videos. For each question, users have four multiple-choice buttons to select the correct answer.
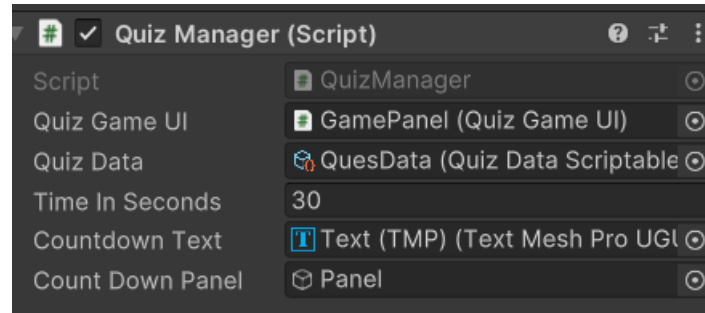
Figure 4.50 Quiz Manager in Unity

The SelectQuestion function plays a pivotal role in the quiz system. It randomly selects questions from a pool of questions using Unity's Random.Range function. Once a question is selected, the SetQuestion function is triggered to present the question to the user. Additionally, the selected question is removed from the available question data to ensure that it is not repeated.

In the Update function, time management is a central aspect. If the game status is set to playing, the timer is decremented by one unit per second. When the timer reaches zero, the game status is transitioned to NEXT, signifying the end of the current game round. Subsequently, the game finish panel is displayed, revealing the user's final score and results.

```
private void SelectQuestion()
{
    //get the random number
    int val = UnityEngine.Random.Range(0, questions.Count);
    //set the selectedQuetion
    selectedQuetion = questions[val];
    //send the question to quizGameUI
    quizGameUI.SetQuestion(selectedQuetion);

    questions.RemoveAt(val);
}

private void Update()
{
    if (gameStatus == GameStatus.PLAYING)
    {
        currentTime -= Time.deltaTime;
        SetTime(currentTime);
    }
}

void SetTime(float value)
{
    TimeSpan time = TimeSpan.FromSeconds(currentTime);                    //set the time value
    quizGameUI.TimerText.text = "Time: " + time.ToString("mm':'ss");   //convert time to Time format

    if (currentTime <= 0)
    {
        //Game Over
        GameEnd();
    }
}
```

Figure 4.51 Quiz Manager Script

The "Answer" function plays a critical role in the quiz system. When a user provides an answer, it is assessed as follows: If the answer is correct, the function adds 50 points to the game score and increments the correct answer count by 1. If the answer is incorrect, the function deducts one life from the remaining life count and triggers the "reduce life" function. If the life count reaches zero, the game status transitions to "NEXT," indicating the end of the current round, and the game finish panel is displayed.

After processing the answer, the system checks whether more questions are available. If there are remaining questions, the game proceeds to the next question. However, if there are no more questions, the game status is updated to "NEXT," and the game finish panel is shown, concluding the quiz.

```csharp
public bool Answer(string selectedOption)
{
    //set default to false
    bool correct = false;
    //if selected answer is similar to the correctAns
    if (selectedQuetion.correctAns == selectedOption)
    {
        //Yes, Ans is correct
        correctAnswerCount++;
        correct = true;
        gameScore += 50;
        quizGameUI.ScoreText.text = "Score:" + "\n" + gameScore;
        quizGameUI.ScoreText2.text = "Score:"+ "\n" + gameScore;
    }
    else
    {
        //No, Ans is wrong
        //Reduce Life
        lifesRemaining--;
        quizGameUI.ReduceLife(lifesRemaining);

        if (lifesRemaining == 0)
        {
            GameEnd();
        }
    }

    if (gameStatus == GameStatus.PLAYING)
    {
        if (questions.Count > 0)
        {
            //call SelectQuestion method again after 1s
            Invoke("SelectQuestion", 0.4f);
        }
        else
        {
            GameEnd();

        }
    }
    //return the value of correct bool
    return correct;
```

Figure 4.52 Quiz Manager Script Answer Function

This class will be storing the questions data, question type and game status.

```
//Datastructure for storeing the quetions data
[System.Serializable]
public class Question
{
    public string questionInfo;         //question text
    public QuestionType questionType;   //type
    public Sprite questionImage;        //image for Image Type
    public AudioClip audioClip;         //audio for audio type
    public UnityEngine.Video.VideoClip videoClip;   //video for video type
    public List<string> options;        //options to select
    public string correctAns;           //correct option
}

[System.Serializable]
public enum QuestionType
{
    TEXT,
    IMAGE,
    AUDIO,
    VIDEO
}

[SerializeField]
public enum GameStatus
{
    PLAYING,
    NEXT
}
```

Figure 4.53 Quiz Manager Script Question Class

In the quiz module, I plan to utilize ScriptableObjects to efficiently store question data. To implement this, I'll create two scripts: one called "Question" to represent individual questions, and another named "QuizDataScriptable" to house a list of questions. This design allows developers to populate the list with various questions.

To enhance convenience, I'll provide customization options such as setting the filename, menu name, and order. This will be achieved using the "CreateAssetMenu" attribute, streamlining the process of working with these assets.

```
[CreateAssetMenu(fileName = "QuestionsData", menuName = "QuestionsData", order = 1)]
public class QuizDataScriptable : ScriptableObject
{
    public List<Question> questions;
}
```

Figure 4.54 Question Data Script
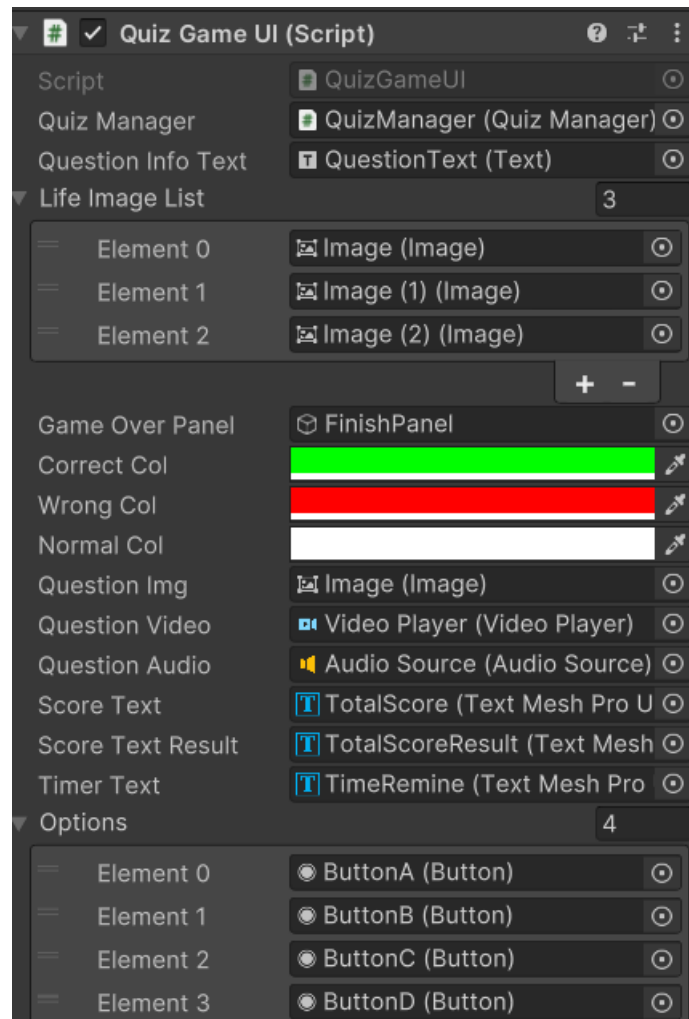
Figure 4.55 Quiz Game UI in Unity

In the QuizGameUI script, the Start function performs two essential tasks. First, it counts the number of buttons present and secondly, it adds event listeners to all the buttons.

The SetQuestions function plays a pivotal role in fetching data from the Question Data file. It then transforms this data into the appropriate question type, which could be text-based, image-based, sound-based, or video-based, depending on the specific question content.

```
private void Start()
{


    //add the listner to all the buttons
    for (int i = 0; i < options.Count; i++)
    {
        Button localBtn = options[i];
        localBtn.onClick.AddListener(() => OnClick(localBtn));
    }

}

/// <summary>
/// <param name="question"></param>
public void SetQuestion(Question question)
{
    //set the question
    this.question = question;
    //check for questionType
    switch (question.questionType)
    {
        case QuestionType.TEXT:
            questionImg.transform.parent.gameObject.SetActive(true);      //activate image holder
            questionVideo.transform.gameObject.SetActive(false);          //deactivate questionVideo
            questionImg.transform.gameObject.SetActive(false);             //activate questionImg
            questionAudio.transform.gameObject.SetActive(false);          //deactivate questionAudio
            break;
```

Figure 4.56 Quiz Game UI Script

The "ReduceLife" function serves to alter the appearance of the life image by changing it to a black color when the user selects the wrong answer. In the case where the question type is audio, the "PlayAudio" function comes into play. This function handles audio playback, allowing the audio to repeat as needed, but it inserts pauses between repetitions, creating intervals before playing the audio again.

```
public void ReduceLife(int remainingLife)
{
    //lifeImageList[remainingLife].color = Color.red;
    lifeImageList[remainingLife].color=Color.black;
}


/// <summary>
/// IEnumerator to repeat the audio after some time
IEnumerator PlayAudio()
{
    //if questionType is audio
    if (question.questionType == QuestionType.AUDIO)
    {
        //PlayOneShot
        questionAudio.PlayOneShot(question.audioClip);
        //wait for few seconds
        yield return new WaitForSeconds(audioLength + 0.5f);
        //play again
        StartCoroutine(PlayAudio());
    }
    else //if questionType is not audio
    {
        //stop the Coroutine
        StopCoroutine(PlayAudio());
        //return null
        yield return null;
    }
}
```

Figure 4.57 Quiz Game UI Script Reduce Life

The "OnClick" function is responsible for managing button behavior in response to user input during the quiz game. When the user selects the correct answer, the function adjusts the button color to indicate correctness and plays the corresponding correct sound. Conversely, if the answer chosen by the user is incorrect, the function modifies the button color to signify incorrectness and triggers the playback of the corresponding wrong sound.

```
void OnClick(Button btn)
{
    if (quizManager.GameStatus == GameStatus.PLAYING)
    {
        //if answered is false
        if (!answered)
        {
            //set answered true
            answered = true;
            //get the bool value
            bool val = quizManager.Answer(btn.name);

            //if its true
            if (val)
            {
                //set color to correct
                btn.image.color = correctCol;
                StartCoroutine(BlinkImg(btn.image));
                source.PlayOneShot(correctClip);
            }
            else
            {
                //else set it to wrong color
                btn.image.color = wrongCol;
                source.PlayOneShot(wrongClip);
            }
        }
    }
}
```

Figure 4.58 Quiz Game UI Script On Click Function

The RetryButton function is designed to reload the current screen or scene, accomplished by utilizing the SceneManager script's loadScene function.

```
public void RestryButton()
{
    SceneManager.LoadScene(SceneManager.GetActiveScene().buildIndex);
}
```

Figure 4.59 Quiz Game UI Retry Function
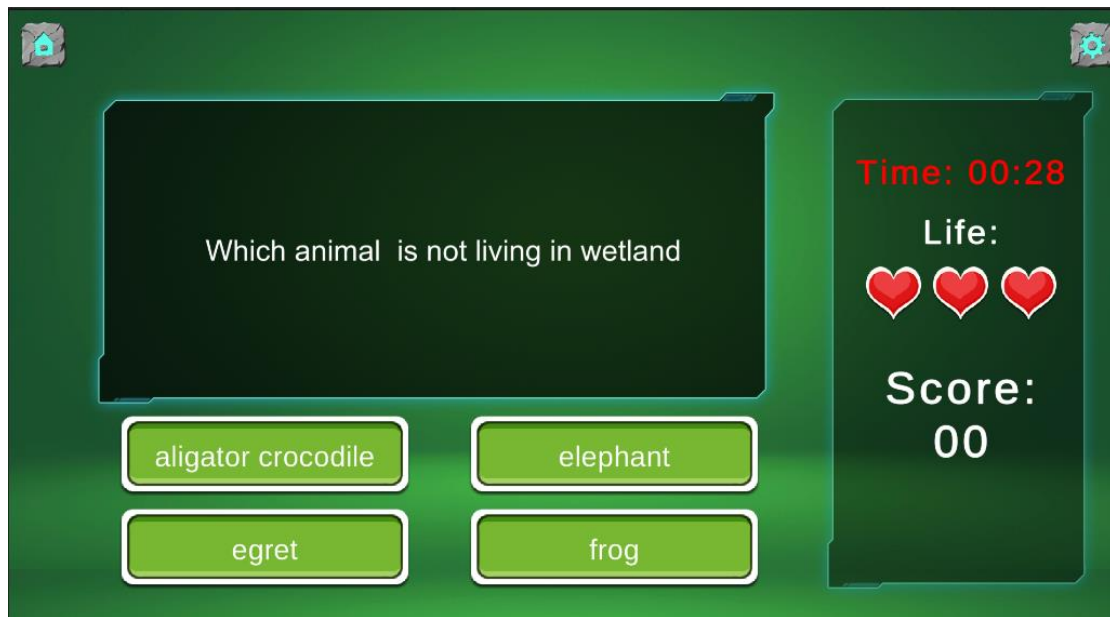
Figure 4.60 Quiz Module Screen Shot

4.2.6 Module 5

Feedback module

The Feedback module is intended to gather user input and opinions about our application. It presents users with multiple questions to solicit their feedback, suggestions, and insights for improvement. The data collected from users will be securely stored and recorded in the database table on phpMyAdmin.

So, the first step is needed to create a database on the phpMyAdmin. After that create a table in the database and define the structure how many attributes inside this table. When the user fill in the feedback form and click the submit button, then the data will store in this database feedback table.

Figure 4.61 phpMyAdmin database feedback table
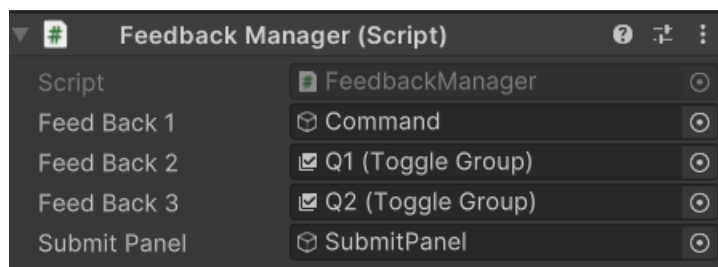
Figure 4.62 Feedback Manager in Unity

Within the FeedbackManager script, three types of feedback questions are implemented. Firstly, "feedback1" is an open-ended question where users can freely provide their comments and feedback after using the application.

Secondly, "feedback2" and "feedback3" are multiple-choice questions presented through a toggle group to capture users' selected choices. The "send" and "post" functions have a crucial role. These functions are responsible for transmitting the collected data to a Google Form. It's essential to create the Google Form first, note down the respective entry numbers for each question, and then incorporate them into the script. When users click the "Submit" button, the "send" and "post" functions are executed, and the data is sent to the specified Google Form, where it can be viewed and analyzed.

```
using MySql.Data.MySqlClient;

public class FeedbackManager : MonoBehaviour
{
    [SerializeField] GameObject FeedBack1;
    [SerializeField] ToggleGroup FeedBack2;
    [SerializeField] ToggleGroup FeedBack3;
    private string feedBack1;
    private string feedBack2;
    private string feedBack3;
    [SerializeField] private GameObject SubmitPanel;

    private string connectionString;
    private MySqlConnection MS_Connection;
    private MySqlCommand MS_Command;
    string query;

    public void sendInfo()
    {
        feedBack1 = FeedBack1.GetComponent<InputField>().text;
        feedBack2 = FeedBack2.GetFirstActiveToggle().transform.GetChild(1).GetComponent<Text>().text;
        feedBack3 = FeedBack3.GetFirstActiveToggle().transform.GetChild(1).GetComponent<Text>().text;

        connection();

        query = "insert into feedback(feedback1, feedback2, feedback3) values( '" + feedBack2 + "' , '" + feedBack3 + "','" + feedBack1 + "');";

        MS_Command = new MySqlCommand(query, MS_Connection);

        MS_Command.ExecuteNonQuery();

        MS_Connection.Close();

        SubmitSuccessful();
    }

    public void connection()
    {

        connectionString = "Server=localhost;Database=fyp_database;User=root;Password=;";
        MS_Connection = new MySqlConnection(connectionString);

        MS_Connection.Open();

    }
}
```

Figure 4.63 Feedback Manager Script
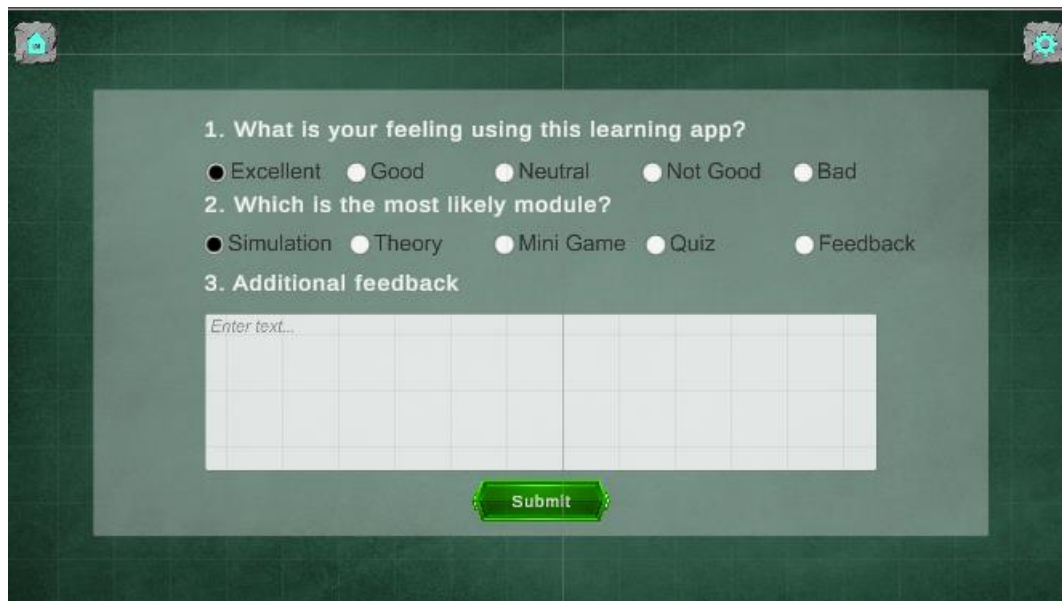
Figure 4.64 Feedback Module Screen Shot

4.2.7 Publishing process

Upon completion of the development phase, the next step involves exporting the project to an executable (exe) file. To achieve this, navigate to the build settings panel,

as illustrated in Figure 4.61, and locate the build button at the bottom of the settings panel. Click on the build button, and after a brief moment, the exe file generation process will commence, allowing for the successful publication of the project.
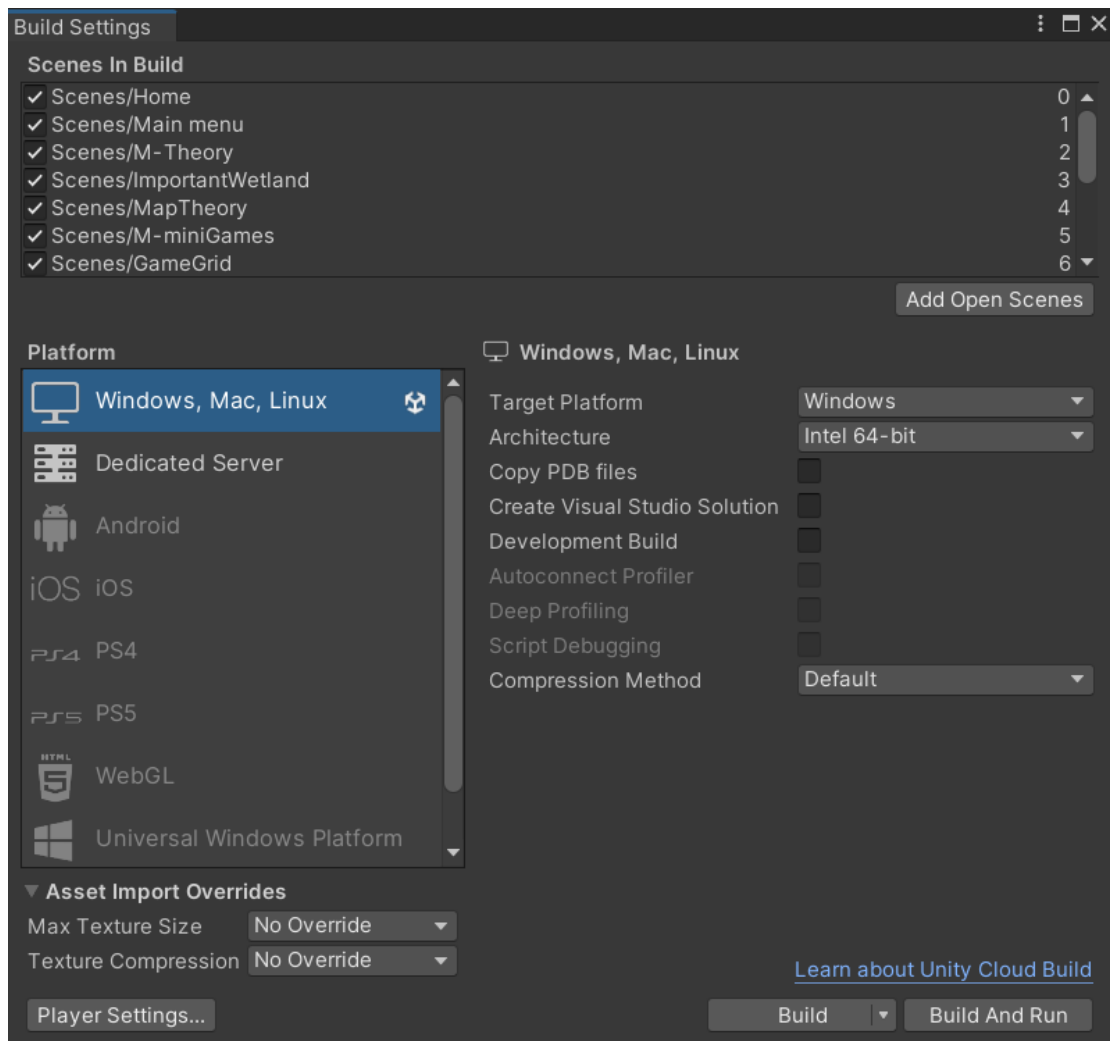
Figure 4.65 Build Setting Screen Shot

CHAPTER 5

TESTING, RESULTS & DISCUSSIONS

5.1 Overview

System testing is a crucial phase in the ADDIE model for instructional design. This phase aims to assess usability and user satisfaction. It ensures that a learning application functions as intended and meets design requirements. The process includes comprehensive planning, creating test cases, and evaluating different aspects of the application. This includes functional testing to verify features, usability testing for user-friendliness, performance testing for responsiveness, compatibility testing across platforms, security testing for data protection, and more. Bugs and issues are tracked and addressed, followed by user acceptance testing and regression testing. Ultimately, system testing ensures a high-quality and user-friendly learning experience. Results, analysis, and discussion will follow this testing phase.

5.2 Method of Testing

The testing phase will involve the use of both SUS and PSSUQ formats. Over the course of one week, more than 30 participants will be engaged in the testing process.

Scale of System Usability (SUS)

For testing usability, the System Usability Scale (SUS) offers a "quick and dirty" but trustworthy method. It comprises of a 10-item questionnaire with five possible responses—Strongly agree to Strongly disagree—given to respondents (System Usability Scale (SUS) | Usability.gov, no date). It enables you to assess a wide range of goods and services, including hardware, software, mobile devices, websites, and applications. John Brooke originally designed it in 1986 (System Usability Scale (SUS) | Usability.gov, no date).

The System Usability Scale is one of the most effective ways to collect statistically accurate data and provide your website with a clear and respectably precise score. It is also quick and inexpensive to start utilising, especially online (Nathan Thomas, no date). SUS is a fairly simple scale to use when giving participants, dependable

outcomes even with tiny sample sizes, and it is reliable because it can distinguish between systems that are useable and those that are not (System Usability Scale (SUS) | Usability.gov, no date). The shocking part is that, although being quicker and less expensive than many other tests, the System Usability Scale is still useful since it consistently achieves its goals and has proven to be reliable and stable (How To Use The System Usability Scale (SUS) To Evaluate The Usability Of Your Website - Usability Geek, no date).

In the project's testing phase, following application development completion, System Usability Scale (SUS) testing is initiated. This process involves creating a Google Form containing SUS template questions to collect user feedback. Target users are recruited through methods like video demonstrations or remote control of the author's computer for a more immersive experience. While the video demonstration method simplifies user participation, the remote-control method, though more complex, offers a more realistic experience. Users provide feedback through the SUS Google Form, enabling the assessment of the application's usability and overall user experience.

Post-Study System Usability Questionnaire (PSSUQ)

In order to conduct more thorough system quality and user satisfaction studies, IBM created the current version of the Post-Study System Usability Questionnaire (PSSUQ) in 1992 (The PSSUQ and usability testing - The Trymata blog, no date). It was developed expressly as a survey model to be used at the conclusion of a usability test, unlike the more well-known SUS. Since user experiences on websites and other digital products are what the questionnaires are specifically designed to measure, they are more accurate. The PSSUQ consists of 16 questions that calculate three key vectors for comprehending a product's usability (The PSSUQ and usability testing - The Trymata blog, no date). These are the vectors: System utility, information excellence and interface excellence.

PSSUQ is used to gauge how usable a website, app, piece of software, system, or other digital product is regarded to be. The questionnaire is longer than most (16 questions), allowing for the capture of very subtle data but also running the risk of potential tester fatigue (The PSSUQ and usability testing - The Trymata blog, no date). You might want to choose a shorter survey model if your test is already quite lengthy.

In any case, PSSUQ can be the right survey for you if you're seeking for a finely calibrated psychometric for precisely and thoroughly analysing web usability (The PSSUQ and usability testing - The Trymata blog, no date).

In the project's testing phase, following application development completion, Post-Study System Usability Questionnaire (PSSUQ) testing is initiated. This process involves creating a Google Form containing PSSUQ template questions to collect user feedback. Target users are recruited through methods like video demonstrations or remote control of the author's computer for a more immersive experience. While the video demonstration method simplifies user participation, the remote-control method, though more complex, offers a more realistic experience. Users provide feedback through the PSSUQ Google Form, enabling the assessment of the application's usability and overall user experience.

5.3 Testing Analysis
Scale of System Usability (SUS)

felt very confident using	I needed to learn a lot of	Total Score
3	2	55
5	2	65
5	1	90
5	1	92.5
5	1	97.5
2	4	70
4	1	67.5
4	4	50
5	2	82.5
5	1	87.5
4	4	62.5
5	1	95
5	1	90
5	1	97.5
5	1	95
4	1	90
5	1	87.5
4	3	70
2	2	50
3	4	55
3	4	55
5	1	100
5	1	100
5	1	95
5	1	95
4	2	70
5	3	65
5	3	65
5	3	65
5	1	100
5	5	60
5	1	77.5
		78.05

Figure 5.1 SUS result and calculation

Here is an overview of the method used in finding your SUS score.

1) For each of the odd numbered questions, subtract 1 from the score.
2) For each of the even numbered questions, subtract their value from 5.
3) Take these new values which you have found and add up the total score. Then multiply this by 2.5.

Per Questions=((B2+D2+F2+H2+J2-5)+(25-C2-E2-G2-I2-K2))*2.5

Total Score=AVERAGE (L2:L33)

Post-Study System Usability Questionnaire (PSSUQ)

Figure 5.2 PSSUQ result and calculation 1

Figure 5.3 PSSUQ result and calculation 2

Figure 5.4 PSSUQ result and calculation 3

The lower the scores, the better the performance and the higher the perceived satisfaction. 1 is "strongly agree" and 7 is "strongly disagree". Scores can be further broken down into (Will T, no date):

- Overall: the average scores of questions 1 to 16
- System Usefulness (SYSUSE): the average scores of questions 1 to 6
- Information Quality (INFOQUAL): the average scores of questions 7 to 12
- Interface Quality (INTERQUAL): the average scores of questions 13 to 15

Per Questions=AVERAGE (B2:B35)

System Usefulness=AVERAGE (B38:G38)

Information Quality=AVERAGE (H38:M38)

Interface Quality=AVERAGE (N38:Q38)

Overall=AVERAGE (G39, M39, Q39)

5.4 Results and Discussions
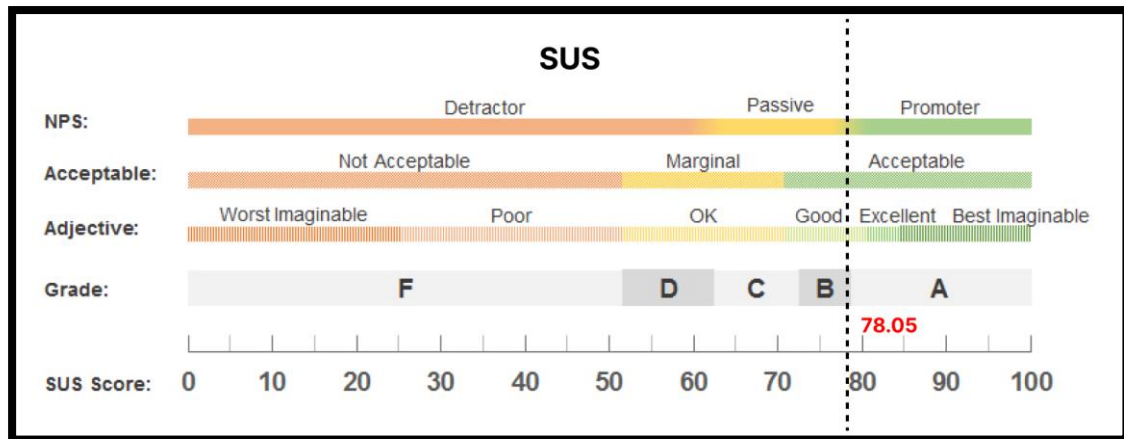
Scale of System Usability (SUS)

Figure 5.5 SUS Testing Graph

The System Usability Scale (SUS) provides a straightforward indicator, either red or green, to gauge the extent of usability enhancements required, without delving into specific issues. The nationwide average SUS score is 68 (How To Use The System Usability Scale (SUS) To Evaluate The Usability Of Your Website - Usability Geek, no date).

In this project, the SUS testing yielded a score of 78, signifying that the project's performance falls into the good range. So, the STS testing result is passive NPS, acceptable and good adjective. While all functions and interfaces are deemed usable, there remains room for improvement. A focus on enhancing user interface and interaction can offer a better learning experience for the target audience. In the context of a high-quality learning application with a 3D virtual environment, enhancing the quality of the virtual environment can lend greater realism and elevate the overall user experience.

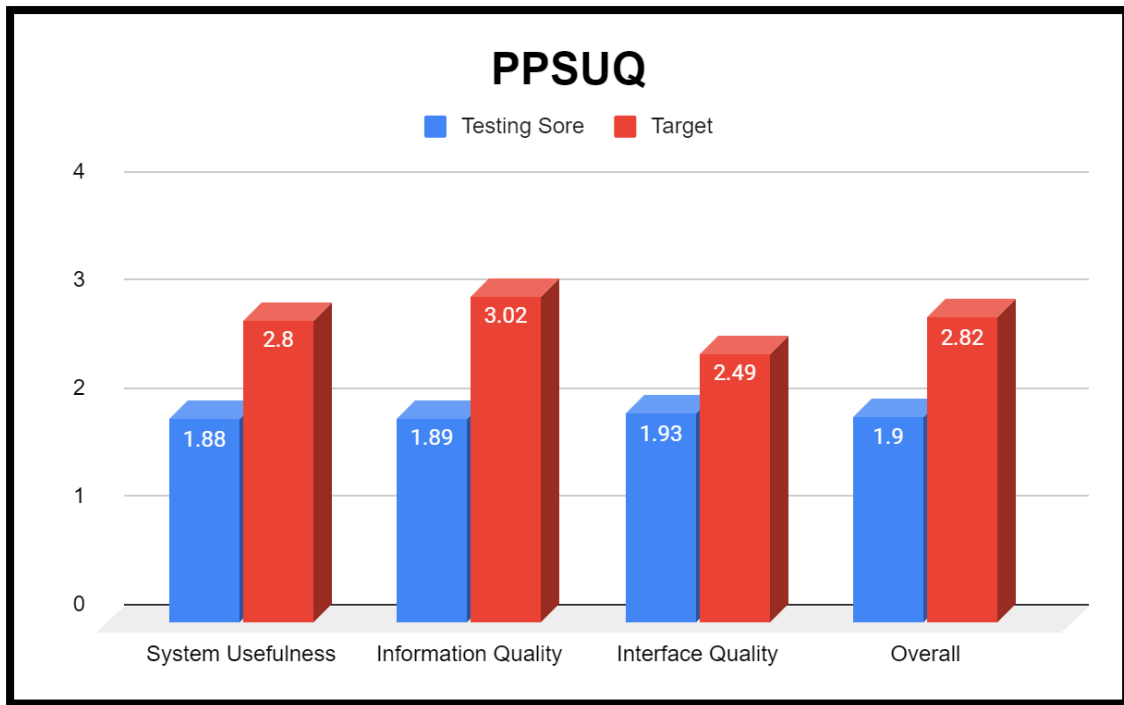**Post-Study System Usability Questionnaire (PSSUQ)**

Figure 5.6 PPSUQ Testing Graph

Numerous academic studies were conducted to evaluate this scale's dependability and establish the average for various structures and specific queries. The following measures established by Sauro and Lewis (2016) using 21 studies and 210 people may be useful in analysing the scores you acquire from your research in order to make sense of the PSSUQ version 3 scores (PSSUQ Calculator - UIUX Trend, no date):

- SYSUSE: 2.80
- INFOQUAL: 3.02
- INTERQUAL: 2.49
- Overall: 2.82

System Usefulness	1.88	2.8
Information Quality	1.89	3.02
Interface Quality	1.93	2.49
Overall	1.9	2.82

Figure 5.7 PPSUQ Result

Based on the provided usability scores for our application, which are 1.88, 1.89, 1.93, and 1.9 show in Figure5.5, it is evident that our application has demonstrated exceptionally high usability. These scores surpass the reference measures established

by Sauro and Lewis (2016) in their analysis of 21 studies and 210 participants. This outcome indicates that our application excels in user-friendliness and efficiency, offering an excellent user experience. These results are a testament to the application's strong usability, and they signify that it effectively caters to user needs and preferences.

CHAPTER 6

CONCLUSION

6.1 Overview

This Chapter provides a comprehensive conclusion to the project, summarizing key findings, challenges encountered, and insights gained during the development of the wetland ecology learning application. It also outlines limitations and offers suggestions for future enhancements. In Section 6.2, the chapter skillfully links research findings to the project's initial problem statements and objectives, showcasing the project's alignment with its educational goals in the realm of wetland ecology. Section 6.3 provides insight into the technical challenges encountered during development, shedding light on the complexities of creating an immersive learning application and highlighting the developer's adaptability and problem-solving skills. Section 6.4 delves into the valuable knowledge and skills acquired throughout the project's journey, emphasizing the growth and professional development experienced in areas such as software development, data analysis, and literature review. Section 6.5 acknowledges the project's limitations, including UI quality, function quality, 3D virtual environment, 3D models, and innovation, and offers insights on addressing these constraints, setting the stage for improvement. Lastly, Section 6.6 charts a forward-looking course by suggesting enhancements for the learning application. It envisions a future with expanded content, emerging technologies, user collaboration, and a vibrant learning community, promising innovation and enriched ecological education.

6.2 Research Findings

This project, "Technology Aided Learning Application for Wetlands Ecology using Constructivism Approach" had been designed and developed according to the project plan to tackle the problem stated in this project. Throughout the project execution, Kanban methodology was practiced and used to schedule the work task. As a result, a system including a Unity Engine and 3D environment web application. In

addition, validation and testing are performed to ensure the system developed meets with requirement specifications and achieves the objectives stated:

1. To study the usage of digital media technologies (2-D and 3-D virtual environments) in aiding the education for wetlands ecology.
2. To design and develop the interactive learning application for wetlands ecology using constructivism approach.
3. To evaluate the effectiveness of the learning application as compared to the traditional linear learning approach.

This project makes several valuable contributions in several aspects. By developing a learning application for wetland ecology, this project contributes to making educational resources more accessible to a broader audience. The online learning platform allows users to engage in immersive learning at their convenience, increasing the accessibility of wetland ecology education.

The project's adoption of a constructivist approach fosters active and experiential learning. The project explores the use of 2-D and 3-D virtual environments as educational tools. By encouraging learners to interact with the material and make connections to their prior knowledge, the project promotes deeper understanding and critical thinking in wetland ecology.

This project collects user data through surveys and tests to understand user satisfaction and expectations for the application. Survey and test results highlight the potential of digital media technologies to enhance the learning experience. 3D modeling provides a more realistic and immersive learning environment that can significantly impact user engagement and understanding.

6.3 Problems Faced

During this project, the author will face some challenges, notably initial exposure to C# programming with limited familiarity with libraries, programming logic, and functions. Unfamiliarity with Unity, encompassing its components, UI design, and file management, presents another learning curve. So during the UI

development need to find the learning material on the internet and practice again and again. The need for high-quality, realistic 3D models, coupled with your inexperience in 3D modeling tools like Blender, underscores the importance of finding or commissioning such models, especially for detailed textures. But most of the 3D model beginners can do the low poly model so the high quality and reality will be challenges. In certain situations, throughout your project, you've encountered the need for image editing, necessitating the acquisition of skills in using Figma. Image editing is a versatile and crucial aspect of various creative and technical projects. Overall, these technical obstacles emphasize the importance of dedicated learning, possibly through tutorials and online resources, and, if feasible, collaborating with individuals or teams possessing expertise in these areas to bolster your project's development.

6.4 Knowledge Gained

Software Development Learning

1. Unity Proficiency: The project required becoming proficient in Unity, a powerful game development engine. This encompassed a deep dive into various aspects, including:

   • UI Design: Understanding how to create and design user interfaces within Unity, ensuring they are visually appealing and functional.

   • Integration with 3D Environments and Models: Learning to seamlessly integrate 3D models and environments within the Unity engine, which is crucial for creating immersive 3D experiences.

   • C# Programming: Developing proficiency in C# programming to implement key functionalities such as library integration, camera control, animation, scene management, and Unity Engine UI development.

2. Blender Mastery: To create realistic 3D models, you embarked on a learning journey with Blender, a popular 3D modeling and animation tool. This involved:

   • 3D Model Editing: Gaining expertise in 3D model editing, including shaping, texturing, and detailing models to meet project requirements.

   • Exporting Models: Learning how to effectively export 3D models from Blender for seamless integration into the Unity environment.

3. Figma for Image Editing and Design: Figma became an essential tool for image editing and design tasks. This included:

- Image Editing: Developing skills in cropping, resizing, enhancing, and optimizing images to align with the project's visual requirements.

- UI and Graphic Design: Creating visually appealing user interfaces and graphics that enhance the overall user experience.

Survey and Testing Data collection and Analysis Learning

4. Data Collection and Analysis: You undertook the challenge of collecting and analyzing data, a crucial aspect of evaluating and improving the project. This encompassed:

- Creating UI Surveys: Developing skills in crafting user-friendly and effective UI surveys to gather user feedback and insights.

- SUS and PPSUQ Testing: Conducting usability testing using the System Usability Scale (SUS) and Post-Study System Usability Questionnaire (PPSUQ) to evaluate the project's usability.

- Data Analysis: Proficiency in data analysis techniques to extract meaningful insights and make data-driven decisions for project enhancement.

Research Learning

5. Research and Literature Review: A critical component involved conducting research and literature review, which included:

- Google Scholar and Mendeley: Gaining competence in using research tools such as Google Scholar and Mendeley for accessing academic literature.

- Citation Management: Learning how to efficiently manage citations and references for comprehensive literature review and proper attribution.

6.5 Limitations

In the pursuit of developing a comprehensive wetland ecology learning application, several notable limitations have emerged. These limitations span various facets of the application's design and functionality. This overview delves into these challenges, highlighting areas that require further attention and improvement to enhance the

learning experience. So, addressing these limitations can pave the way for a more enriching and engaging educational platform.

The application faces challenges related to content depth and variety. While it aims to educate users about wetland ecology, there's a need for more in-depth information and a broader range of educational resources. This learning application just talks about a little bit of wetland environment ecosystem, some theory, animals, and plant information, and the relationship between animals, plants, and the environment, so the learning material is not detailed and wide range.

The user experience encompasses several aspects, including device compatibility and UI quality. Ensuring compatibility across a multitude of devices, including mobile devices and tablets, is vital for widespread accessibility. But this learning application only can support the computer-base so this is one of the limitations. Simultaneously, UI quality improvement plays a pivotal role in enhancing the user experience. At the same time, the UI still can improve a lot, the main topic color, background image, and the button design.

The application's immersive elements, including the 3D virtual environment, 3D models, and innovative features, present opportunities for enhancement. This learning application already includes the 3D virtual environment and 3D models but the limitation is that if the 3D resource file is too large then the hardware requirement will not user user-friendly. So in this application, the 3D virtual environment and 3D model will reduce many details and the environment area and model type and number will be less.

6.6 Future Enhancement

For future enhancements, consider a roadmap that focuses on continuous improvement. Expand the application's content by incorporating additional wetland ecosystems, species, and ecological scenarios to offer a more comprehensive learning experience.

The application faces challenges related to content depth and variety. To provide a comprehensive understanding, it should delve deeper into the subject matter, offering a wealth of detailed content that fosters a robust learning experience. This entails not only expanding the information available but also diversifying the content to cater to various aspects of wetland ecology.

The user experience encompasses several aspects, including device compatibility and UI quality. Ensuring compatibility across a multitude of devices, including mobile devices and tablets, is vital for widespread accessibility. Optimizing the application's design for diverse screen sizes enhances its usability. Simultaneously, UI quality improvement plays a pivotal role in enhancing the user experience. A more visually appealing, intuitive, and user-friendly interface can significantly boost user engagement and navigation. These aspects collectively contribute to a seamless and enjoyable learning experience.

The application's immersive elements, including the 3D virtual environment, 3D models, and innovative features, present opportunities for enhancement. Expanding the 3D virtual environment in both size and realism is critical for providing a deeper and more immersive learning experience. Diversifying the library of 3D models and elevating their animations and details can make the virtual ecosystem more captivating. Additionally, introducing innovative elements, such as gamification, augmented reality (AR), or virtual reality (VR), can inject interactivity and freshness into the learning experience.

REFERENCES

Allen, W.C. (2006) 'Overview and Evolution of the ADDIE Training System', *http://dx.doi.org/10.1177/1523422306292942*, 8(4), pp. 430–441. Available at: https://doi.org/10.1177/1523422306292942.

Arkün, S. and Akkoyunlu, B. (2008) 'A Study on the development process of a multimedia learning environment according to the ADDIE model and students' opinions of the multimedia learning environment', *Interactive Educational Multimedia*, 17, pp. 1–19. Available at: https://revistes.ub.edu/index.php/IEM/article/view/11902 (Accessed: 4 April 2023).

Bassi, N. *et al.* (2014) 'Status of wetlands in India: A review of extent, ecosystem benefits, threats and management strategies', *Journal of Hydrology: Regional Studies*, 2, pp. 1–19. Available at: https://doi.org/10.1016/J.EJRH.2014.07.001.

Claireyuke (2021) *Ecosystem Simulation by claireyuke*. Available at: https://claireyuke.itch.io/ecosystem-simulation (Accessed: 20 April 2023).

Debbie G. (2014) *iBiome-Wetland Review for Teachers | Common Sense Education*. Available at: https://www.commonsense.org/education/reviews/ibiome-wetland (Accessed: 28 February 2023).

Eoghan Quigley (no date) *ADDIE: 5 Steps To Effective Training Courses | LearnUpon*. Available at: https://www.learnupon.com/blog/addie-5-steps/ (Accessed: 28 March 2023).

Hedden, M.K. *et al.* (2017) 'Teaching Sustainability Using an Active Learning Constructivist Approach: Discipline-Specific Case Studies in Higher Education', *Sustainability 2017, Vol. 9, Page 1320*, 9(8), p. 1320. Available at: https://doi.org/10.3390/SU9081320.

Nathan Thomas (no date) *How To Use The System Usability Scale (SUS) To Evaluate The Usability Of Your Website - Usability Geek*. Available at: https://usabilitygeek.com/how-to-use-the-system-usability-scale-sus-to-evaluate-the-usability-of-your-website/ (Accessed: 1 September 2023).

Imed Bouchrika, P. (2022) *The ADDIE Model Explained: Evolution, Steps, and Applications | Research.com*. Available at: https://research.com/education/the-addie-model (Accessed: 28 March 2023).

Kim, J.S. (2005) 'The effects of a constructivist teaching approach on student academic achievement, self-concept, and learning strategies', *Asia Pacific Education Review*, 6(1), pp. 7–19. Available at: https://doi.org/10.1007/BF03024963.

Kurt, S. (2017) *Definitions of The Addie Model - Educational Technology*. Available at: https://educationaltechnology.net/definitions-addie-model/ (Accessed: 4 April 2023).

LEPAYA PETER (2021) *Linear vs Non-linear Learning and the Future of Work*. Available at: https://lepaya.com/en/linear-and-non-linear-learning/ (Accessed: 28 February 2023).

Li, Y.W. (2016) 'Transforming Conventional Teaching Classroom to Learner-Centred Teaching Classroom Using Multimedia-Mediated Learning Module', *International Journal of Information and Education Technology*, 6(2), pp. 105–112. Available at: https://doi.org/10.7763/IJIET.2016.V6.667.

Maureen Tam (2015) *(PDF) Constructivism, Instructional Design, and Technology: Implications for Transforming Distance Learning*. Available at: https://www.researchgate.net/publication/26391080_Constructivism_Instructional_Design_and_Technology_Implications_for_Transforming_Distance_Learning (Accessed: 17 April 2023).

Mitsch, W.J., Bernal, B. and Hernandez, M.E. (2015) 'Ecosystem services of wetlands', *https://doi.org/10.1080/21513732.2015.1006250*, 11(1), pp. 1–4. Available at: https://doi.org/10.1080/21513732.2015.1006250.

Multimedia Introduction (no date). Available at: https://www.tutorialspoint.com/multimedia/multimedia_introduction.htm (Accessed: 19 April 2023).

Muruganantham Ganesan (2015) *(PDF) Developing of E-content pcakge by using ADDIE Model*. Available at: https://www.researchgate.net/publication/339102976_Developing_of_E-content_pcakge_by_using_ADDIE_Model (Accessed: 4 April 2023).

Nobel Laureate Carl Wieman (2002) *Natural Selection - Mutation | Genetics | Selection - PhET Interactive Simulations*. Available at: https://phet.colorado.edu/en/simulations/natural-selection (Accessed: 20 April 2023).

Olusegun, S. (no date) 'Constructivism Learning Theory: A Paradigm for Teaching and Learning', 5(6), pp. 66–70. Available at: https://doi.org/10.9790/7388-05616670.

Ou, K.L., Chu, S.T. and Tarng, W. (2021) 'Development of a Virtual Wetland Ecological System Using VR 360° Panoramic Technology for Environmental Education', *Land 2021, Vol. 10, Page 829*, 10(8), p. 829. Available at: https://doi.org/10.3390/LAND10080829.

PBS Kids (2015) *Games : Jungle Jeopardy . PLUM LANDING | PBS KIDS*. Available at: https://pbskids.org/plumlanding/games/ecosystem/jungle_jeopardy.html (Accessed: 20 April 2023).

Phungsuk, R., Viriyavejakul, C. and Ratanaolarn, T. (2017) 'Development of a problem-based learning model via a virtual learning environment', *Kasetsart Journal of Social Sciences*, 38(3), pp. 297–306. Available at: https://doi.org/10.1016/J.KJSS.2017.01.001.

Will T (no date) *PSSUQ Calculator - UIUX Trend*. Available at: https://uiuxtrend.com/pssuq-calculator/ (Accessed: 1 September 2023).

Sebelas, U. and Rejekiningsih, T. (2020) 'European Journal of Educational Research Development of Interactive Multimedia Learning Courseware to Strengthen Students' Character An-nisa Nur Sholihah Indah Septiani *'. Available at: https://doi.org/10.12973/eu-jer.9.3.1267.

Seifan, M., Dada, D. and Berenjian, A. (2019) 'The effect of virtual field trip as an introductory tool for an engineering real field trip', *Education for Chemical Engineers*, 27, pp. 6–11. Available at: https://doi.org/10.1016/J.ECE.2018.11.005.

Spring Bay Studio (no date) *iBiome-Wetland: School Edition | Environmental Game | Spring Bay Studio*. Available at: https://springbaystudio.com/iBiome-Wetland/ (Accessed: 28 February 2023).

Stemler, L.K. (1997) 'Educational Characteristics of Multimedia: A Literature Review', *Jl. of Educational Multimedia and Hypermedia*, 6(4), p. 359.

System Usability Scale (SUS) | Usability.gov (no date). Available at: https://www.usability.gov/how-to-and-tools/methods/system-usability-scale.html (Accessed: 1 September 2023).

The PSSUQ and usability testing - The Trymata blog (no date). Available at: https://trymata.com/blog/learn/pssuq/ (Accessed: 1 September 2023).

ThinMatrix (2018) *Steam 上 的 Equilinox*. Available at: https://store.steampowered.com/app/853550/Equilinox/ (Accessed: 28 February 2023).

Wan Abdul Rahim Wan Mohd Isa, A.I.H.S.N.N.N.A.O. (2019) '3D Virtual Learning Environment'. Available at: https://doi.org/10.35940/ijeat.F1015.0986S319.

Wang, J. (2020) 'The Design of 3D-Virtual Learning Environments in the View of System Theory*'.

Appendix A: Questionnaire Google Form

The survey of FYP : Learning Application for Wetlands Ecology

Dear participants,

My name is Ngwee Jia Ler. I am Year 3 student of Bachelor of Software Engineering (Honors) from University Tunku Abdul Rahman, Sungai Long Campus.

I am conducting my Final Year Project (FYP) from January 2023 to September 2023. The main purpose of this survey is to know what is the significance of learning applications and **understand the user requirement and their expectation** for Learning Application for Wetlands Ecology.

The questionnaire will take around 5-10 minutes to be completed.

Your responses are much appreciated.

For further inquiries, feel free contact at **WhatsApp 018-3184712** or email me through **ngweejialer@1utar.my**.

ngweejialer@1utar.my （未分享） 切换帐号

*必填

Gender *

○ Male

◉ Female

Age *

○ 18-30

○ 31-40

○ 41-50

○ 51-60

下一页   清除表单内容

Q1. Using learning applications can enhance your understanding and knowledge. *

	1	2	3	4	5	
Strongly Disagree	◯	◯	◯	◯	◯	Strongly Agree

Q2. What is the reason for using the learning application? *

- ☐ Acquire new knowledge or skills
- ☐ Increase existing knowledge or skills
- ☐ Easy to understand
- ☐ Revision and prepare for a specific exam or certification
- ☐ Free time to study
- ☐ 其他: _____

Q3. What type of digital simulation you are prefer to use? *

- ☐ 3D virtual environment interactive simulation
- ☐ 2D virtual environment interactive simulation
- ☐ Graphic simulation
- ☐ Video simulation
- ☐ 其他: _____

Q4. The utilization of a 3D virtual environment can provide a highly interactive, *
realistic, and effective approach to learn about wetland ecosystems.

	1	2	3	4	5	
Strongly Disagree	◯	◯	◯	◯	◯	Strongly Agree

Q5. Which of the following features would you interested in wetlands simulation module? Choose at least two. *

☐ Change the weather patterns

☐ Change the season type

☐ Background music and sound effects

☐ Human activities(pollution and habitat destruction)

☐ Map

☐ 其他: _____

Q6. More interactive will spend more time to explore the learning application. What is the longest time you can accept on study. *

○ 30 min

○ 1 hours

○ 2 hours

○ 3 hours above

Q7. Mini games allow users to learn and play together, helping to motivate users to keep learning. *

	1	2	3	4	5	
Strongly Disagree	○	○	○	○	○	Strongly Agree

Q8. Which of the following type of game would you prefer in mini games module? Choose at least two. *

- [ ] Puzzle Game
- [ ] Spelling Game
- [ ] Match Game
- [ ] Food Chain Game
- [ ] 其他:

Q9. Learning assessment attract user in the wetland ecosystem learning application. *

	1	2	3	4	5	
Strongly Disagree	◯	◯	◯	◯	◯	Strongly Agree

Q10. Which of the following type of questions would you prefer in assessment module? Choose at least two. *

- [ ] Multiple choice questions
- [ ] Fill in the blank questions
- [ ] Short Answer questions
- [ ] True False questions
- [ ] 其他:

Q11. In assessment module, which of the following type of feedback would you like to received after answering the questions. *

☐ Give the correct answer

☐ Explanation of the correct answer

☐ Marking(Scores)

☐ 其他: _____

Q12. Theory-base is the most direct and effective way to increase knowledge in learning. *

	1	2	3	4	5	
Strongly Disagree	○	○	○	○	○	Strongly Agree

Q13. Which of the following type of material would you prefer in theory-base module? Choose at least two. *

☐ Text-base material

☐ Video-base material

☐ Graphic-base material

☐ Audio-base material

☐ Animation-base material

☐ 其他: _____

Q14. Feedback module is one of the important features to improve learning application. *

	1	2	3	4	5	
Strongly Disagree	○	○	○	○	○	Strongly Agree

Q15. Which of the following type of function would you prefer in feedback module. *

☐ Feedback function

☐ Social media follow

☐ FQA

☐ 其他:

Q16. Please share any suggestions and improvements about the module (Interface design, functionality, features, innovations) would you expect in wetlands ecosystem learning application.

您的回答

上一页　提交　　　清除表单内容

Appendix B: Testing Form Result

Scale of System Usability (SUS)

I think that I would like to use this system frequently
31 responses
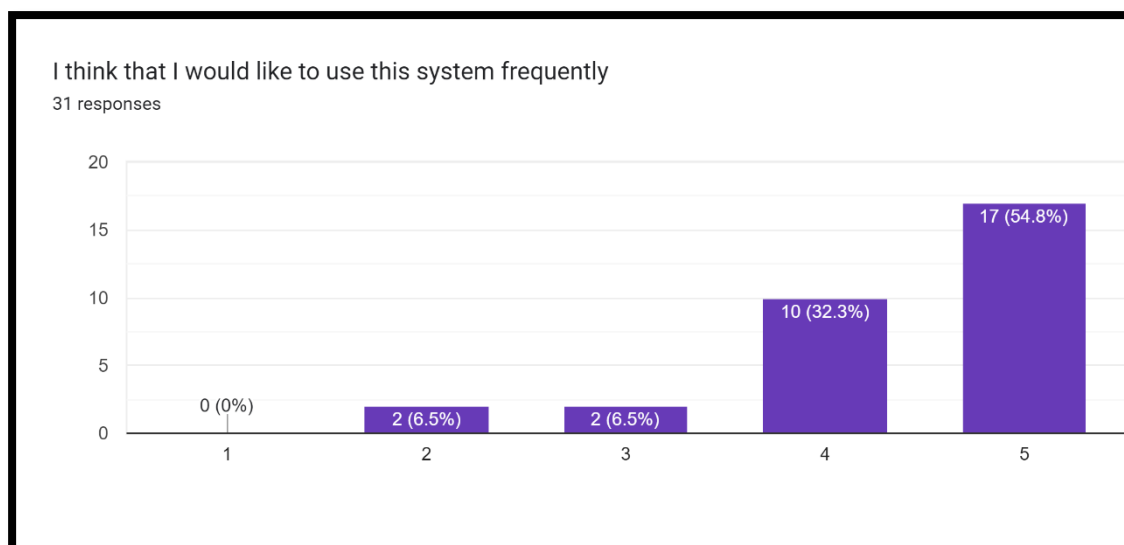
I found the system unnecessarily complex.

31 responses
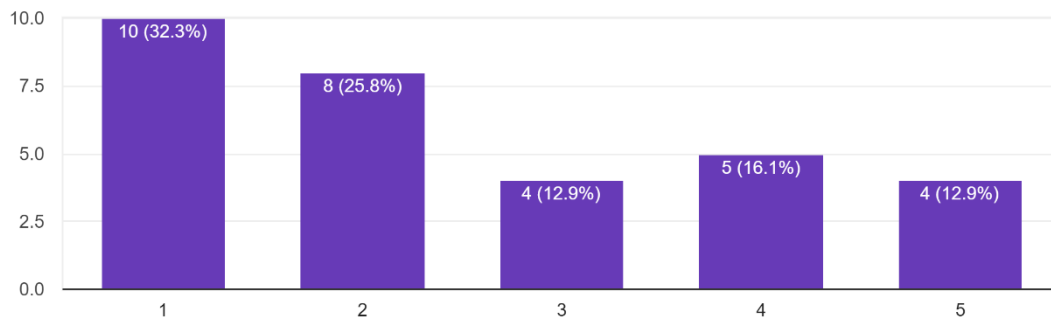
I thought the system was easy to use.

31 responses
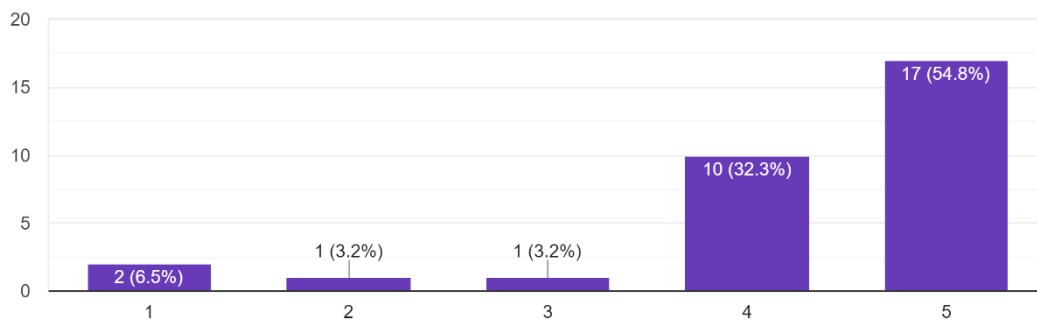
I think that I would need the support of a technical person to be able to use this system.

31 responses
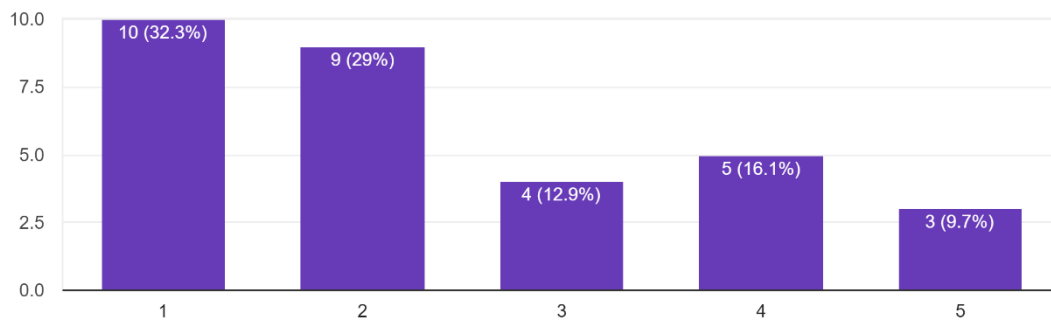
I found the various functions in this system were well integrated.
31 responses
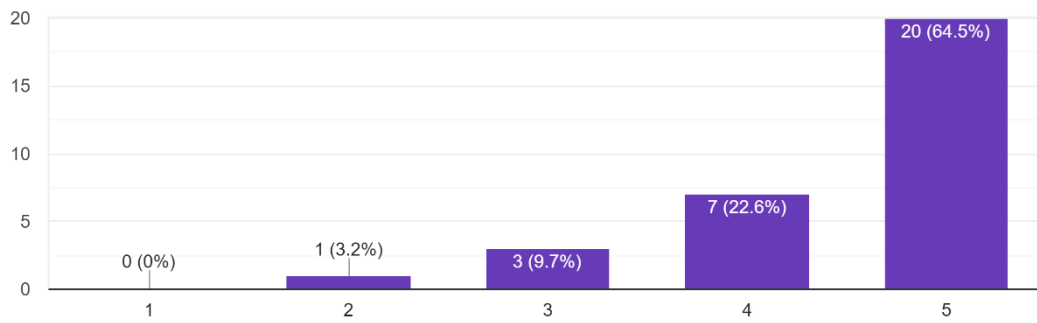
I thought there was too much inconsistency in this system.
31 responses
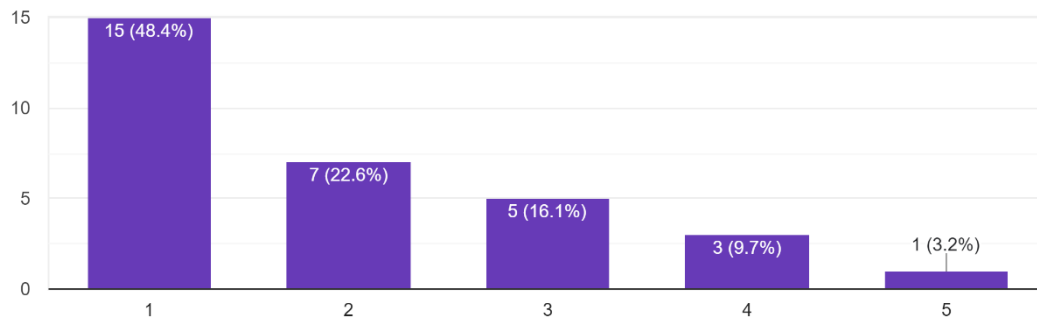
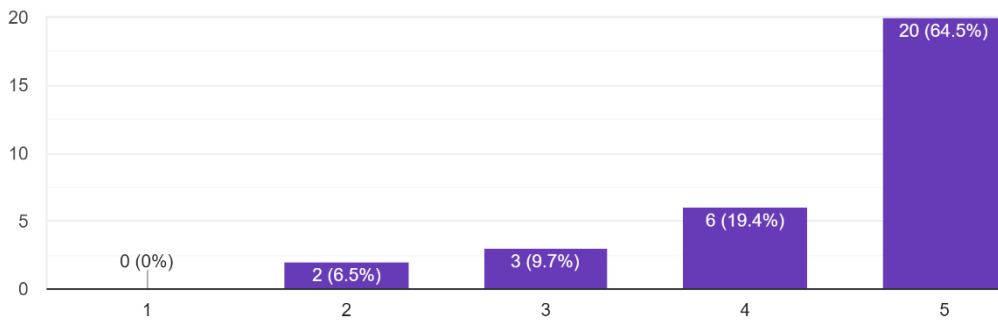## I would imagine that most people would learn to use this system very quickly.
31 responses
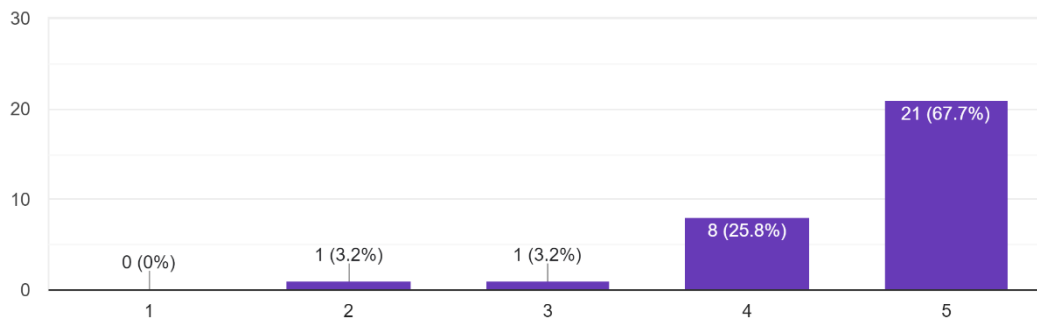
I found the system very cumbersome to use.
31 responses

Rating	Count	Percentage
1	13	41.9%
2	11	35.5%
3	3	9.7%
4	4	12.9%
5	0	0%
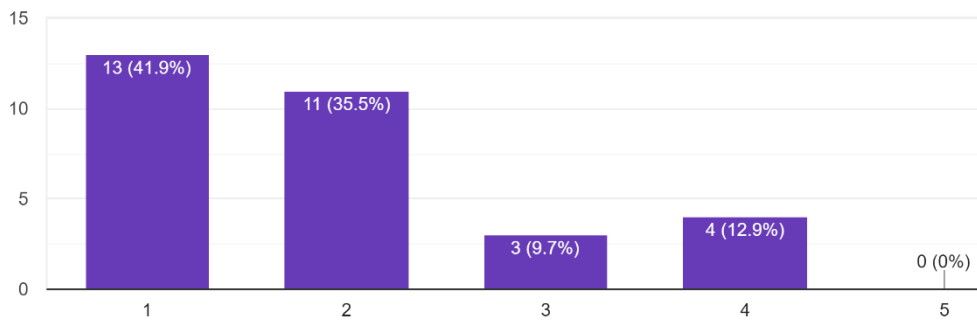
I felt very confident using the system.
31 responses

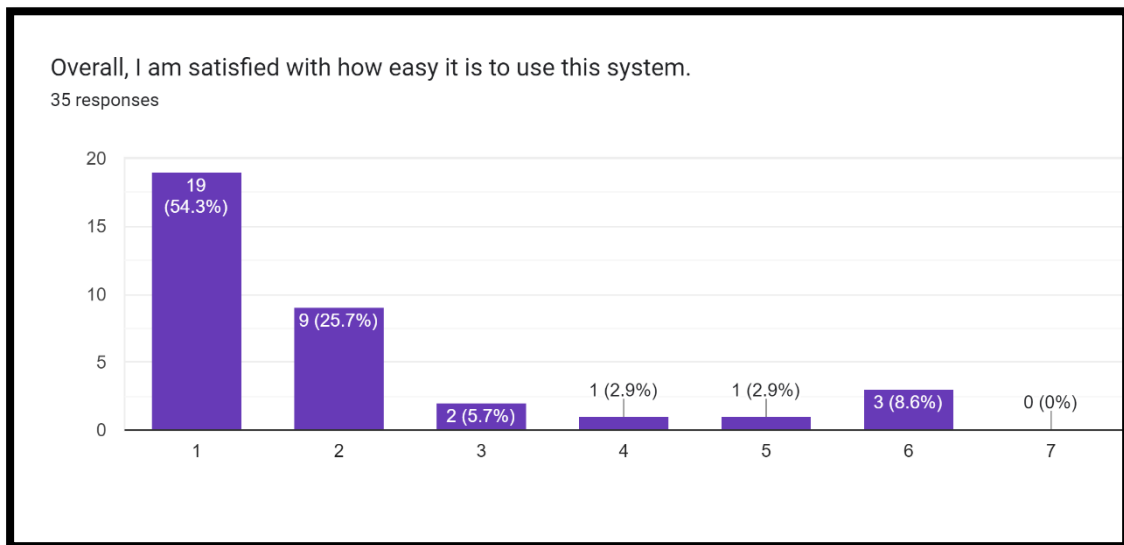| Rating	Count	Percentage
1	0	0%
2	2	6.5%
3	3	9.7%
4	6	19.4%
5	20	64.5%

I needed to learn a lot of things before I could get going with this system.
31 responses

Rating	Count	Percentage
1	16	51.6%
2	5	16.1%
3	4	12.9%
4	5	16.1%
5	1	3.2%

Post-Study System Usability Questionnaire (PSSUQ)

Overall, I am satisfied with how easy it is to use this system.

35 responses

It was simple to use this system.

35 responses

I was able to complete the tasks and scenarios quickly using this system.
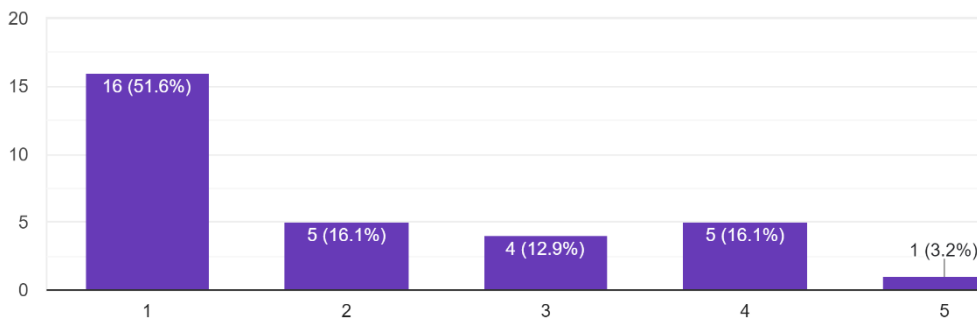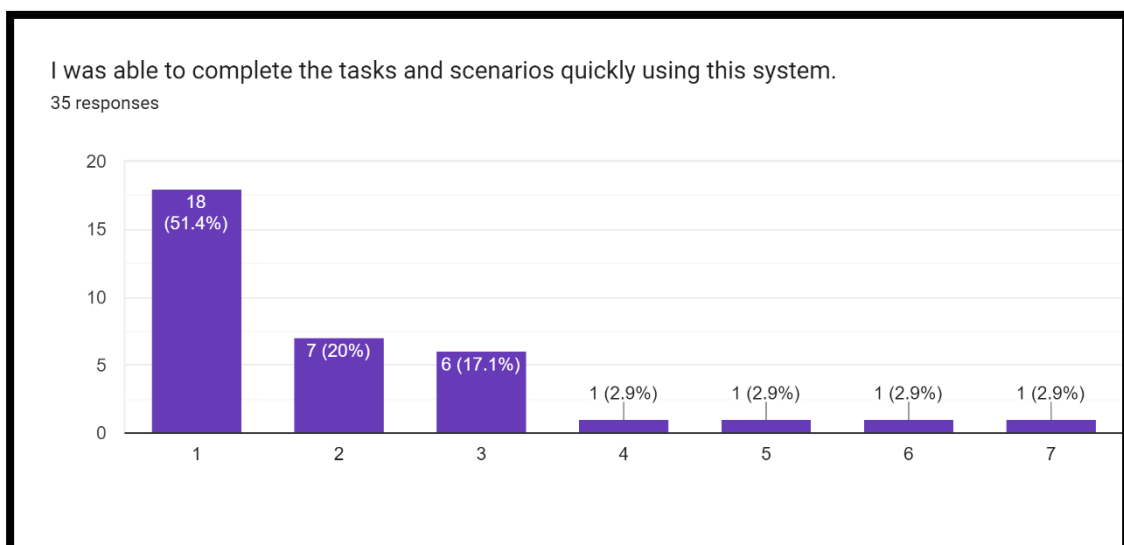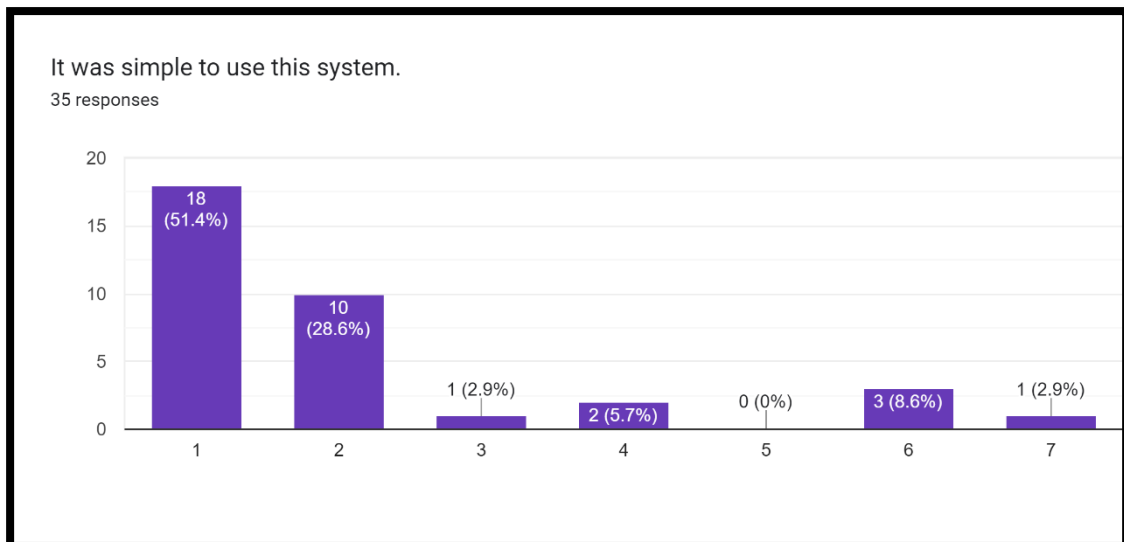
35 responses

I felt comfortable using this system.
35 responses

Bar chart (scale 1 to 7):
- 1: 20 (57.1%)
- 2: 9 (25.7%)
- 3: 1 (2.9%)
- 4: 2 (5.7%)
- 5: 1 (2.9%)
- 6: 1 (2.9%)
- 7: 1 (2.9%)
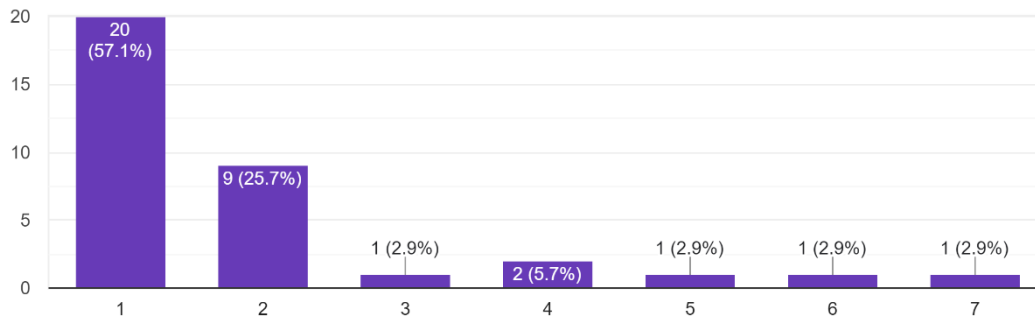
It was easy to learn to use this system.
35 responses

Bar chart (scale 1 to 7):
- 1: 20 (57.1%)
- 2: 10 (28.6%)
- 3: 1 (2.9%)
- 4: 0 (0%)
- 5: 0 (0%)
- 6: 3 (8.6%)
- 7: 1 (2.9%)
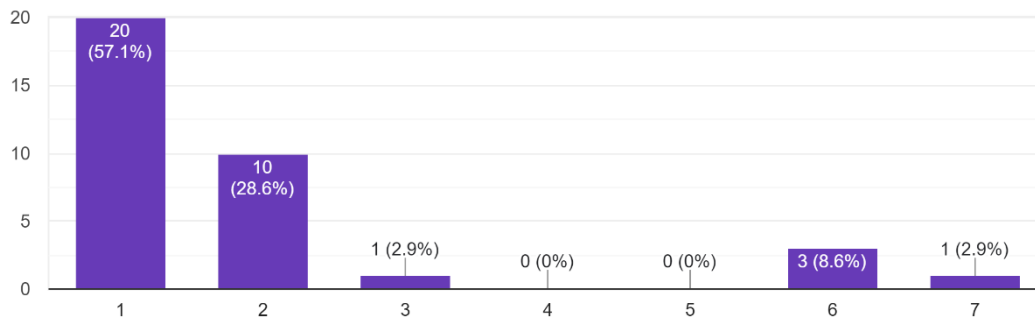
I believe I could become productive quickly using this system.
35 responses

Bar chart (scale 1 to 7):
- 1: 18 (51.4%)
- 2: 9 (25.7%)
- 3: 4 (11.4%)
- 4: 0 (0%)
- 5: 1 (2.9%)
- 6: 1 (2.9%)
- 7: 2 (5.7%)
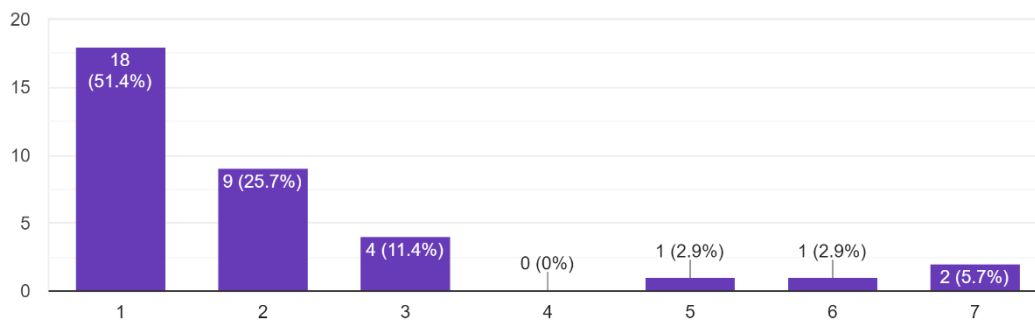
The system gave error messages that clearly told me how to fix problems.

35 responses

Bar chart:
- 1: 13 (37.1%)
- 2: 12 (34.3%)
- 3: 5 (14.3%)
- 4: 1 (2.9%)
- 5: 0 (0%)
- 6: 3 (8.6%)
- 7: 1 (2.9%)
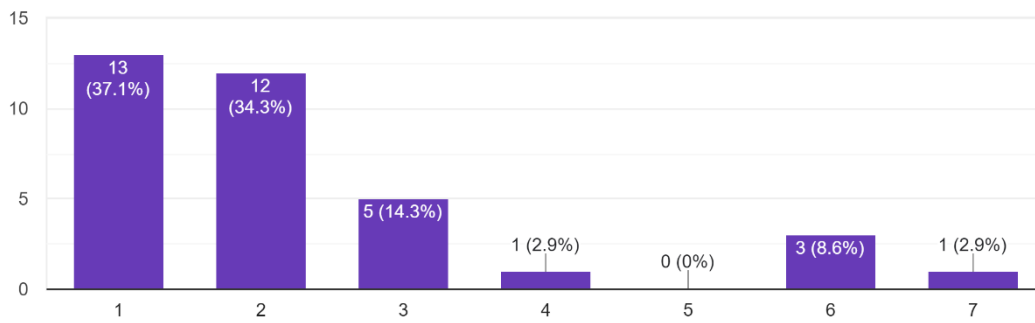
Whenever I made a mistake using the system, I could recover easily and quickly.

35 responses

Bar chart:
- 1: 20 (57.1%)
- 2: 8 (22.9%)
- 3: 3 (8.6%)
- 4: 0 (0%)
- 5: 1 (2.9%)
- 6: 2 (5.7%)
- 7: 1 (2.9%)
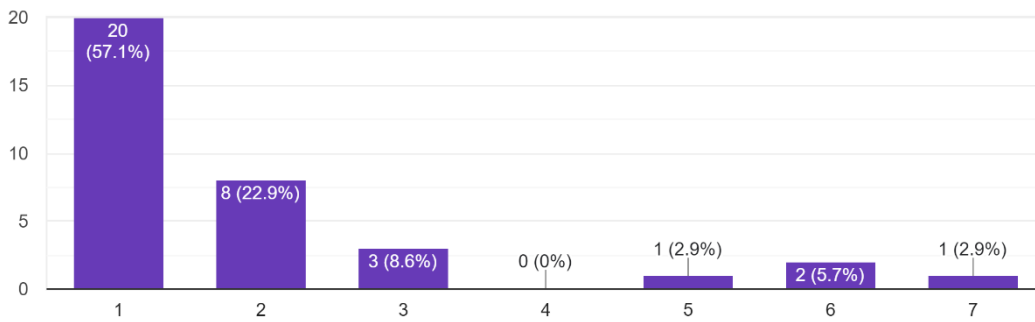
The information (such as online help, on-screen messages, and other documentation) provided with this system was clear.

35 responses

Bar chart:
- 1: 18 (51.4%)
- 2: 10 (28.6%)
- 3: 2 (5.7%)
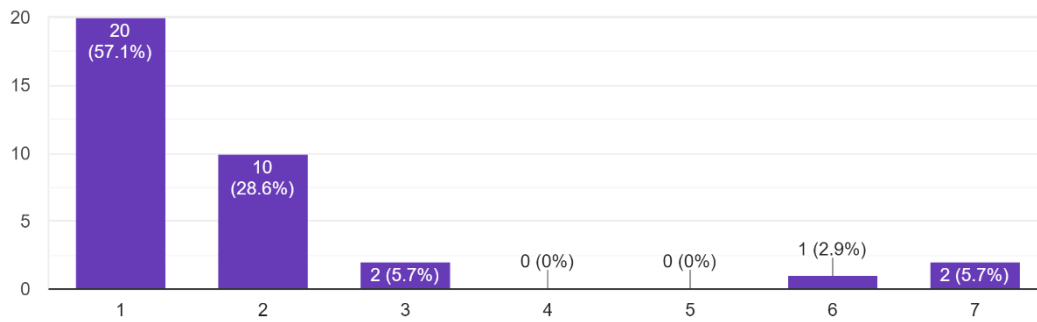- 4: 1 (2.9%)
- 5: 2 (5.7%)
- 6: 1 (2.9%)
- 7: 1 (2.9%)
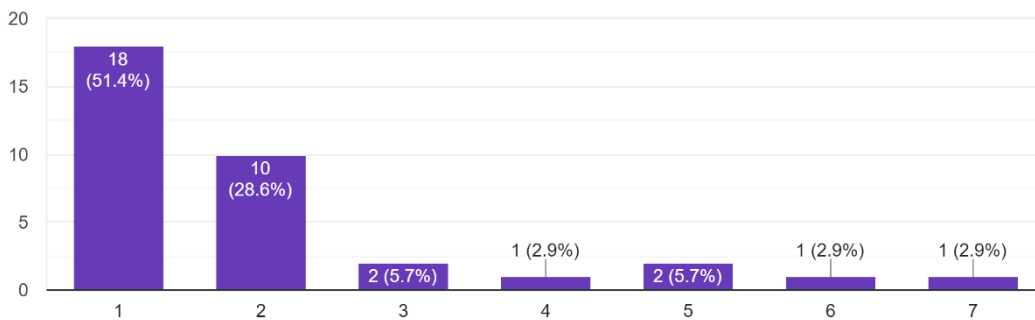
It was easy to find the information I needed.
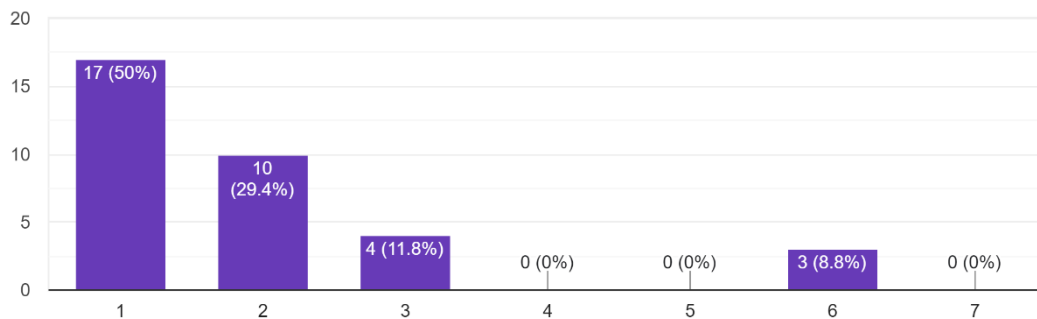35 responses
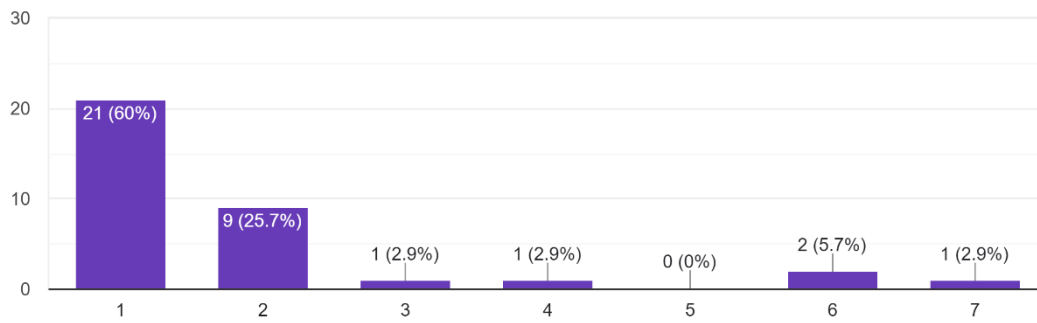
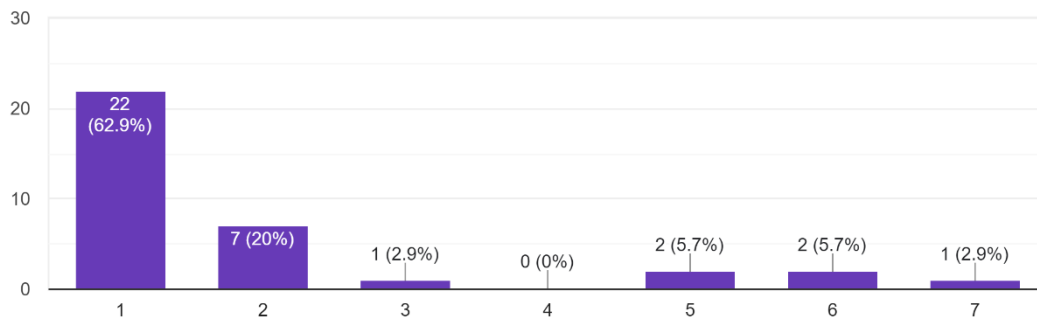**The information was effective in helping me complete the tasks & scenarios.**
35 responses

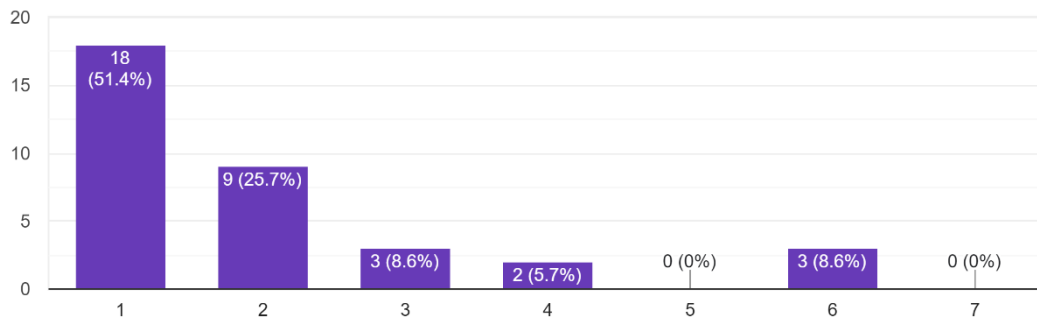**The organization of information on the system screens was clear.**
34 responses

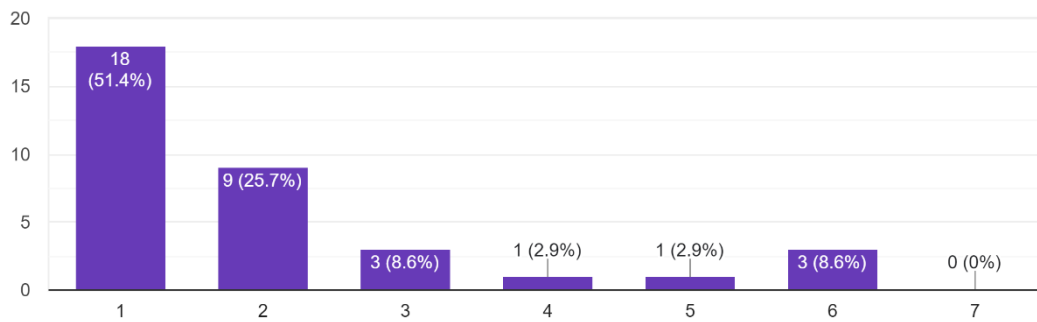**The interface of this system was pleasant.**
35 responses

I liked using the interface of this system.
35 responses

This system has all the functions and capabilities I expect it to have.
35 responses

Overall, I am satisfied with this system.

35 responses