**Creating an Educational Game in Virtual Reality**

Dovydas Simoliunas[[1]](#footnote-1)

*School of Computing Science, Newcastle University, UK*

**Abstract**

This paper contains research on educational games set in virtual reality. It starts off by providing a definition, based on descriptions from peer-reviewed articles, of what makes a good educational game. Then we use the Unity game engine to create a base for a marine life educational game set in virtual reality. We compare our game against a list of common factors shared by educational games. These factors are then used to measure the game’s validity as an educational VR game and to underline any of its shortcomings.

*Keywords: Virtual Reality, VR, Educational game, Data-oriented design pattern, Unity Game Engine, C#*

# Introduction

## Context

The purpose of an average video game is to provide its users with a fun and entertaining experience. The interactivity that traditional video games provide in their gameplay could easily be considered educational as it often requires problem solving skills. However, while many games require some levels of skill and possess learning curves in their gameplay, they often are not considered educational. Educational games are defined as serious games with the primary purpose of education. [Add stuff, this doesn’t make sense]

Virtual reality provides an extra layer of immersion to video games. The first-person perspective with the action-based camera movements allows the player to feel like they’re experiencing the gameplay themselves, instead of just the character.

## Problem

The main reason behind this project is that the Newcastle University’s Marine Biology department has asked us to make an educational game about marine life. This prompted us to research educational games in Virtual Reality and design one of our own. However, the resources given for this project were not enough to make a complete educational game in the time frame given. The total time constraint for this project was roughly two months. This would allow us to only create a small-scale game using a pre-made game engine, like the Unity game engine [1]. Furthermore, the lack of available free-to-use assets posed another restriction on the game development, as it is difficult to make a full game without appropriate assets. These two restrictions had caused us to make a compromise and redefine our goals for this project.

## Rationale

It was decided to create a data-oriented framework that would allow its users to quickly set up game scenes. The users could simply choose the assets that are going to appear in the game by filling out a text file. For this model an underwater scene worked best, with various sea creatures and other game components being spawned around the player. The next step was to make an example game using this framework with the assets that are available.

This paper will describe the process of making the VR game and will discuss what makes a good educational game in virtual reality based on descriptions from peer-reviewed sources.

## Aim and Objectives

Aim – To create an Educational VR game about marine life adhering to best practices of educational VR games.

Objectives:

1. Research qualities of good educational games.
2. Research best practices of VR games.
3. Write a data-oriented framework for easily creating simple game scenes.
4. Create an educational example game using that framework and any available assets.
5. Attempt to objectively judge the game against the research results.

# Background

# Design

The game was designed in a simple way. There is a single start scene that has a menu that teleports the player to an underwater scene. This menu showcases a selection of different underwater scene layouts that can be loaded in. These layouts can be one of two types – learning and test.

The learning type scenes have various interactable sea creatures swimming around the player. The player can move about the scene looking at the creatures. To help with the immersion, some of the creatures can be grabbed onto and they take the player with them as they swim around. The creatures can also be clicked on with the VR controllers. This stops the animal’s AI and opens a UI pop-up window with educational information about the selected animal. Releasing the clicked button hides the UI and reenables the object’s AI for it to continue moving. Once the player is finished exploring, they can return to the start scene and select a new one.

The test type scene is also underwater with multiple-choice test UI in front of the player. In this type of scene, to avoid the user moving out of the test area, the locomotion system is turned off and the animals cannot be grabbed onto.

Due to not having sufficient underwater assets, the best approach to the problem was to use a data-oriented design pattern. It allowed us to make the game with limited assets as more could easily be loaded into the scenes when acquired. Furthermore, using this pattern to make a simple string table would allow the user to load in custom information that is displayed about each creature.

The underwater scene was designed by covering the camera with a transparent dark blue screen. [Expand here]

# Development

Unity game engine provides the XR interaction toolkit package that was used for this project. It has made the development of this game much simpler by providing pre-set ways to translate VR actions into in-game events.

Most of the data that is being loaded in this project is in a JSON format. This is mostly due to Unity having a simple to use JsonUtility library that reads and writes data to and from serializable objects. Data formatted this way carries the benefit of informing any editor about what data to put in them as they contain the variable names as the key to each value field.

Initially, most of the development time had been spent on setting up and learning to use the OpenXR Unity package [reference here].

# Evaluation

# Conclusions

# Further work

Audio messages on the info

Limit the player from escaping the scene bounds

Colliders on the camera

More interaction with the animals.

Different types of tests (spelling, image)

Bottom of the ocean scene

Switching to Scriptable objects in Unity.

# Acknowledgements

# References

1. <https://unity.com/> (Last Accessed: 18-05-2023)

1. d.simoliunas@newcastle.ac.uk [↑](#footnote-ref-1)