

# Creating 3D games with Unity3D

1<sup>st</sup> Boris Fuchs

*Mobile Computing*

*University of Applied Sciences Upper Austria*

Hagenberg, Austria

boris.fuchs@students.fh-hagenberg.at

2<sup>nd</sup> David Mitterlehner

*Mobile Computing*

*University of Applied Sciences Upper Austria*

Hagenberg, Austria

david.mitterlehner@students.fh-hagenberg.at

3<sup>rd</sup> Anika Seibezeder

*Mobile Computing*

*University of Applied Sciences Upper Austria*

Hagenberg, Austria

anika.seibezeder@students.fh-hagenberg.at

**Abstract**—This document is a model and instructions for  $\text{\LaTeX}$ . This and the `IEEEtran.cls` file define the components of your paper [title, text, heads, etc.]. **\*CRITICAL: Do Not Use Symbols, Special Characters, Footnotes, or Math in Paper Title or Abstract.**

**Index Terms**—component, formatting, style, styling, insert

## I. INTRODUCTION

Nowadays creating realistic 3D games is getting a lot easier, due to various tools like Unity3D or the Unreal Engine. Game developers need little to no programming skills to make games in a minimum amount of time, because a great amount of premade assets and scripts are available for a very low cost. Most of the developing work can be done in the graphical editor, like creating 3D figures and adding properties, which can include self made scripts or physical properties. The majority of the existing game editors provide developing on multiple platforms including Smartphones and the most common game consoles. However Unity3D is better suited for devices with low end graphics cards, because it is more lightweight than the Unreal Engine. Compared to it, the available assets of Unity3D are not as high quality, but more economical on memory and graphics.

The main points of this paper are: outlining the workflow of the game development process in Unity3D which consists of creating an environment, making a Third-Person-Character and some enemies, and comparing the basic functionality and simplicity of Unity3D and the Unreal Engine. The advantages and disadvantages of both editors are examined and demonstrated.

The first point of the game development process in Unity3D is to create an environment, with a small forest, some mountains and a few cottages. Then some textures are added to the components. Secondly the Third-Person-Character is linked with a script, so it can move around in the environment. The third point is to create some enemies, that are chasing the player. Lastly animations for both, the character and the enemies, are provided and added to the game.

The goal of this paper is to demonstrate the simplicity of Unity3D by creating a small 3D game in Unity3D. To keep the costs low, only freely available assets are used. The focus is not on scripting, but on the features of the graphical editor provided by Unity3D and the Unreal Engine.

## II. CREATING AN ENVIRONMENT

### A. Subsection 1

The `IEEEtran` class file is used to format your paper and style the text. All margins, column widths, line spaces, and text fonts are prescribed; please do not alter them. You may note peculiarities. For example, the head margin measures proportionately more than is customary. This measurement and others are deliberate, using specifications that anticipate your paper as one part of the entire proceedings, and not as an independent document. Please do not revise any of the current designations.

## III. CHARACTER CONTROLS

Before you begin to format your paper, first write and save the content as a separate text file. Complete all content and organizational editing before formatting.

Keep your text and graphic files separate until after the text has been formatted and styled. Do not number text heads— $\text{\LaTeX}$  will do that for you.

### A. Subsection 1

Define abbreviations and acronyms the first time they are used in the text, even after they have been defined in the abstract. Abbreviations such as IEEE, SI, MKS, CGS, ac, dc, and rms do not have to be defined. Do not use abbreviations in the title or heads unless they are unavoidable.

Before you begin to format your paper, first write and save the content as a separate text file. Complete all content and organizational editing before formatting.

Keep your text and graphic files separate until after the text has been formatted and styled. Do not number text heads— $\text{\LaTeX}$  will do that for you.

#### IV. ADDING ANIMATIONS

Define abbreviations and acronyms the first time they are used in the text, even after they have been defined in the abstract. Abbreviations such as IEEE, SI, MKS, CGS, ac, dc, and rms do not have to be defined. Do not use abbreviations in the title or heads unless they are unavoidable.

#### V. COMPARING UNITY3D WITH THE UNREAL ENGINE

Define abbreviations and acronyms the first time they are used in the text, even after they have been defined in the abstract. Abbreviations such as IEEE, SI, MKS, CGS, ac, dc, and rms do not have to be defined. Do not use abbreviations in the title or heads unless they are unavoidable.

#### REFERENCES

- [1] G. Eason, B. Noble, and I. N. Sneddon, "On certain integrals of Lipschitz-Hankel type involving products of Bessel functions," *Phil. Trans. Roy. Soc. London*, vol. A247, pp. 529–551, April 1955.
- [2] J. Clerk Maxwell, *A Treatise on Electricity and Magnetism*, 3rd ed., vol. 2. Oxford: Clarendon, 1892, pp.68–73.
- [3] I. S. Jacobs and C. P. Bean, "Fine particles, thin films and exchange anisotropy," in *Magnetism*, vol. III, G. T. Rado and H. Suhl, Eds. New York: Academic, 1963, pp. 271–350.
- [4] K. Elissa, "Title of paper if known," unpublished.
- [5] R. Nicole, "Title of paper with only first word capitalized," *J. Name Stand. Abbrev.*, in press.
- [6] Y. Yorozu, M. Hirano, K. Oka, and Y. Tagawa, "Electron spectroscopy studies on magneto-optical media and plastic substrate interface," *IEEE Transl. J. Magn. Japan*, vol. 2, pp. 740–741, August 1987 [Digests 9th Annual Conf. Magnetism Japan, p. 301, 1982].
- [7] M. Young, *The Technical Writer's Handbook*. Mill Valley, CA: University Science, 1989.