Journal of Advanced Zoology

ISSN: 0253-7214
*Volume **44** Issue **S-6 Year 2023** Page **918:922***

Educational and Immersive Game Development

***K S Adarsh*[1], Gururaj Suresh Baby[2], *Harish Kumar*[3], *Hruthik C K*[4], *Dr. Venkatesh Prasad*[5]**

[1,2,3,4]BTech Computer Science and Engineering, REVA University Bangalore, India

**Corresponding author's: K S Adarsh*

Article History

Received: 06 June 2023
Revised: 05 Sept 2023
Accepted: 30 Nov 2023

Abstract

Educational Systems have remained the same for decades while the technology and its effect on our daily lives has undergone tremendous progress. Technology has been incorporated in various other fields with great success while education has been conservative and slow to adapt, this has been problematic in a number of ways, since entertainment, news, social media have evolved to better suit the human psyche. This creates a gap between these aforementioned systems and education where education is way behind in terms of engagement and immersion compared to entertainment and such. We propose one way to tackle this divide is to use "Educational Games" which use technology to make education engaging and entertaining.

Keywords: *Video Games, Educational Games, Warfare, Tactics, Technology, Engagement.*

1. Introduction

The motivation to play, and therefore to learn, that might be provided by digital educational games teases researchers and developers. Current educational games often fail in their attempt to compete with commercial games and to provide successful learning. Often some learning is added to digital games or some gameplay is added to educational applications. Successful educational games, however, require merging professional game design with sound pedagogical strategies, creating a new hybrid format. Moreover, a methodology is required that allows continuously balancing gaming and learning challenges and the learner's abilities and knowledge in order to retain an immersive gaming experience. In this article we introduce approaches to game design and didactic design, as well as a framework for adaptive interventions in educational games

Problem Definition:

a. Educational Games:

Firstly, to make the game interesting enough to hold the student's attention.

Secondly, to educate him in a manner that he/she doesn't feel like its studying i.e., the educational part of the program should almost feel like a natural byproduct.

b. Voice Control:

To recognize the command said by the player,

Then correctly identify it and look up the appropriate response stored in a file or within the program itself.

And finally send the response to the game with minimum latency possible to ensure smooth, immersive gameplay.

2. Materials And Methods

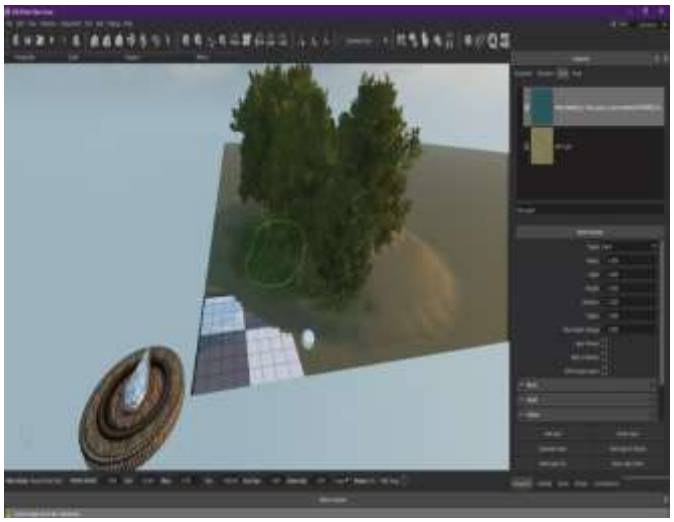
We use the "mount and blade 2: bannerlord mod kit", which is an internal development engine of Taleworlds™.

Using this system instead of traditional game engines provides us with a number of asset files and generation tools which are incredibly useful and efficient.

We create battle scenes, weapons, soldiers, Maps and use the taleworlds Artificial intelligence to simulate battles.

- The data structure of the entire system of game files is handled using xml,
- Where each file stored like a scene, character entity, texture, etc. are referenced in a different submodule for each type of data all based on a hierarchical system.
- The processing of all the systems is internally done using c# code.

The voice commands shall be implemented using apis from Microsoft, Google, etc. whichever is found suitable



Generating Maps In The Game Engine

CODE

A. Voice Recognition Code (Python3):

```
# import required module
import speech_recognition as sr
# explicit function to take input commands
# and recognize them
def takeCommandHindi():
    r = sr.Recognizer()
    with sr.Microphone() as source:
        # seconds of non-speaking audio before
        # a phrase is considered complete
        print('Listening')
        r.pause_threshold = 0.7
        audio = r.listen(source)
        try:
            print("Recognizing")
            Query = r.recognize_google(audio, language='hi-In')
            # for listening the command in indian english
            print ("the query is printed=", Query, """)
        # handling the exception, so that assistant can
        # Ask for telling again the command
        except Exception as e:
```

```
        print(e)
        print("Say that again sir")
        return "None"
    return Query
# Driver Code
# call the function
if(takeCommandHindi()=='आक्रमण'):
    import keyboard
    import time
    time.sleep(2)
    keyboard.press_and_release('f1')
    time.sleep(1)
    keyboard.press_and_release('f3')
```

A. Game Engine Code (Xml):

```
<?xml version="1.0" encoding="utf-8"?>
<Module>
  <Name value = "CustomBattle"/>
  <Id value = "CustomBattle"/>
  <Version value = "e1.5.9"/>
  <DefaultModule value="true"/>
  <SingleplayerModule value="true"/>
 <Official value="true" />
 <DependedModules>
    <DependedModule Id="Native" DependentVersion="e1.5.9"/>
    <DependedModule Id="SandBoxCore" DependentVersion="e1.5.9"/>
  </DependedModules>
  <SubModules>
    <SubModule>
      <Name value = "CustomBattleSubModule"/>
      <DLLName value = "TaleWorlds.MountAndBlade.CustomBattle.dll"/>
      <SubModuleClassType value = "TaleWorlds.MountAndBlade.CustomBattle.CustomBattleSubModule"/>
      <Tags>
        <Tag key="DedicatedServerType" value ="none" />
      </Tags>
    </SubModule>
  </SubModules>
  <Xmls>
    <XmlNode>
      <XmlName id="NPCCharacters" path="custombattlecharacters"/>
      <IncludedGameTypes>
```

```
            <GameType value = "CustomGame"/>
            <GameType value = "EditorGame"/>
          </IncludedGameTypes>
        </XmlNode>
        <XmlNode>
          <XmlName id="Scene" path="custom_battle_scenes"/>
        </XmlNode>
    </Xmls>
</Module>
```

B. Game Module Code (C#)

```
<Module>
    <Name value="Example Mod"/>
    <Id value="ExampleMod"/>
    <Version value="v1.0.0"/>
    <SingleplayerModule value="true"/>
    <MultiplayerModule value="false"/>
    <DependedModules>
        <DependedModule Id="Native"/>
        <DependedModule Id="SandBoxCore"/>
        <DependedModule Id="Sandbox"/>
        <DependedModule Id="CustomBattle"/>
        <DependedModule Id="StoryMode" />
    </DependedModules>
    <SubModules>
        <SubModule>
            <Name value="ExampleMod"/>
            <DLLName value="ExampleMod.dll"/>
            <SubModuleClassType value="ExampleMod.MySubModule"/>
            <Tags>
                <Tag key="DedicatedServerType" value="none" />
                <Tag key="IsNoRenderModeElement" value="false" />
            </Tags>
        </SubModule>
    </SubModules>
    <Xmls/>
 </Module>
```

SYSTEM REQUIREMENTS

Basic system with a dedicated GPU

Higher end graphics card suggested for much smoother gameplay and high resolutions such as 4K

Directx latest version

Microphone

TOOLS USED

- Mount And Blade Bannerlord Mod Kit – Engine
- Blender – 3D Animation and Rigging
- Google Speech Recognition API – Voice Recognition
- Multiple Python Libraries – PyAudio, pipwin, Keyboard
- Microsoft Visual C++ – Editing C# and XML codes

APPLICATIONS

We believe this format of education could be applied to almost any field including biology, chemistry, physics, mathematics, programming, language learning, etc. In fact, there has already been a few educational game systems proving this for example coolmathgames.com and some educational games which are used to teach preschoolers. But we suggest this be adopted in all levels of the system and with varied types of games.

In Game Render.

4. Conclusion

The Game shall provide engaging educational content with immersive environments, We aim to add numerous battles to choose from including mythical battles from “Mahabharata” and “Ramayana”. Future works including fleshing out a Indian subcontinent map with interesting and important scenes. We are actively in progress to integrate speech-to-text module for seamless conversational command structure which allows commanding in regional languages or historically accurate battle chants.

References:

1.Dondlinger, Mary. (2007). Educational Video Game Design: A Review of the Literature. Journal of Applied Educational Technology. 4. Alec Radford, Jeffrey Wu,Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, “Language models are unsupervised multi task learners”,Semantic Scholars, Corpus ID: 160025533, 2018.

2.Blumberg, F. C., Almonte, D. E., Barkhardori, Y., & Leno, A. (2014). Academic lessons from video game learning. In F. C. Blumberg (Ed.), Learning by playing: Video gaming in education (p. 3–12). Oxford University Press.

3. Marina Papastergiou, Digital Game-Based Learning in high school Computer Science education: Impact on educational effectiveness and student motivation, Computers & Education, Volume 52, Issue 1, 2009, Pages 1-12, ISSN 0360-1315

4.Pablo Moreno-Ger, Daniel Burgos, Iván Martínez-Ortiz, José Luis Sierra, Baltasar Fernández-Manjón, Educational game design for online education, Computers in Human Behavior, Volume 24, Issue 6, 2008, Pages 2530-2540, ISSN 0747-5632

5.De Freitas, Sara. "Are Games Effective Learning Tools? A Review of Educational Games." Journal of Educational Technology & Society 21, no. 2 (2018): 74-84. Accessed April 9, 2021. http://www.jstor.org/stable/26388380