Subject: UltimateOpenGL [BETA] 3D engine
Posted by Xemuth on Mon, 18 Nov 2019 14:45:30 GMT
View Forum Message <> Reply to Message

Hello community,

Today seems to be the good day to present my project of doing 3D game engine named UltimateOpenGL.
The project is still at beta state but I think it's good enough to actually show it and show some code and example.
Moreover having some return on the way of how it works can be a good point to improve it.

So let's dive inside :

UltimateOpenGL is a 3D motor used to make 3D games made by a beginner. It's designed to be "code only". it mean you don't have (at this time) any GUI to help you create scene with your mouse.
At this time, UOGL allow Scene creation, insertion of multiple 'GameObject'(Shapes, Lights), insertion of multiple camera per scene. Model loading and rendering with Assimp

https://github.com/Xemuth/UltimateOpenGL_V3

Here is some screenshot of some example I made :

here you have some code example of how it's working :

```
    //***All basic code to set Up glwf is coming before see my exemple or Readme of my GITHUB**///
    context.AddTexture("sand",TransformFilePath("/Textures/sand.jpg")); //Load sand texture

    Scene& myScene = context.AddScene("main");
    myScene.AddCamera("main");
    myScene.SetBackGroundColor(context.TransformRGBToFloatColor(40,180,200));

    Mesh m; //Use to simulate the flat floot under the model
    m.GetVertices().Add().SetPosition(glm::vec3(-20.0f, 0.0f, -20.0f)).SetTexCoords(glm::vec2(0.0f, 20.0f));
    m.GetVertices().Add().SetPosition(glm::vec3(20.0f, 0.0f, -20.0f)).SetTexCoords(glm::vec2(20.0f, 20.0f));
    m.GetVertices().Add().SetPosition(glm::vec3(20.0f, 0.0f,  20.0f)).SetTexCoords(glm::vec2(20.0f, 0.0f));
    m.GetVertices().Add().SetPosition(glm::vec3(20.0f, 0.0f,  20.0f)).SetTexCoords(glm::vec2(20.0f, 0.0f));
    m.GetVertices().Add().SetPosition(glm::vec3(-20.0f, 0.0f,  20.0f)).SetTexCoords(glm::vec2(0.0f, 0.0));
```

```
    m.GetVertices().Add().SetPosition(glm::vec3(-20.0f, -0.0f,
-20.0f)).SetTexCoords(glm::vec2(0.0f, 20.0f));

    Object3D& floor =  myScene.CreateGameObject<Object3D>("floor",m); //Lets create floor
    floor.BindTexture("sand");

 Object3D& modele =  myScene.CreateGameObject<Object3D>("modele");
 modele.LoadModel("C:\\Upp\\myapps\\ExempleUltimateOpenGL_V3\\obj upp\\upp.obj");
//Loading of model

 modele.GetTransform().SetNewPosition(glm::vec3(0,4,0)); //Set new position
 modele.GetTransform().ScaleNewValue(glm::vec3(0.05f,0.05f,0.05f)); //Scale the model

 modele.SetOnDrawFunction([](GameObject& gm){ //Bind event on draw
  double rotation = glm::cos(context.GetEllapsedTime())/100;
  gm.GetTransform().RotateFromEulerAngles(context.GetDeltaTime() * 2,glm::vec3(0,1,0) );
 });

 myScene.Load();
    while(!glfwWindowShouldClose(window)) {
        processInput(window);

  glfwSetWindowTitle(window, "UltimateOpenGL V3 - " +AsString(context.GetFPS()) +" FPS");
  try{
   context.Draw();  //Draw the context
  }catch(UGLException& e){
   LOG(e.what());
  }

     glfwSwapBuffers(window);
     glfwPollEvents();
 }
 glfwTerminate();
```

I will not dive deeply in this example by the time. But if you are interested, you can find more information and some code example in thoses both github page :
 UltimateOpenGL
Exemple using it

Lot of feature are coming, Animation, Physics, Sound...

Let me know in coment what you though about it :)