

**FACULTATEA: Automatica si Calculatoare**

**SPECIALIZAREA: Calculatoare si Tehnologia Informatiei**

**DISCIPLINA: Prelucrare grafică**

**PROIECT: Fragment of the household of a house OpenGL Scene**

|  |  |
| --- | --- |
| **Prof. coordonator:**  **Grigor Sonia** | **Student:**  **Hirean Roxana-Maria**  **Grupa: 30234** |

# Cuprins

**1. Prezentarea temei alese** ..................................................................................................................... 3

1. **Scenariul** ............................................................................................................................................ 3
   1. **Descrierea scenei si a obiectelor** .................................................................................................. 4
   2. **Functionalitati** .............................................................................................................................. 5
2. **Detalii de implementare** .................................................................................................................... 7
   1. **Functii si algoritmi** ....................................................................................................................... 7
   2. **Modelul grafic** ............................................................................................................................ 10
   3. **Structuri de date** ........................................................................................................................ 11
   4. **Ierarhia de clase** ........................................................................................................................ 12
3. **Prezentarea interfeței grafice utilizator / manual de utilizare** ....................................................... 12
4. **Concluzii și dezvoltări ulterioare** .................................................................................................... 13
5. **Referințe** .......................................................................................................................................... 13

**1. Prezentarea temei alese**

Tema acestui proiect consta in realizarea unei scene 3D, cat mai realista si apropiata de detaliile vietii de zi cu zi. Scena pe care am ales sa o modelez este un fragment din gospodaria unei case. Am urmarit in principal calitatea texturarii imaginilor, aranjarea obiectelor in scena si calitatea randarii scenei(toate sa fie cat mai realiste).

Scopul acestui proiect a fost invatarea principiilor din OpenGL si familiarizarea cu utilizarea practica a acestora in mediul de programare Visual Studio, utilizand librariile prezentate la laborator (OpenGl, GLFW, GLM, etc.), de asemenea utilizarea mediului de modelare 3D Blender. Utilizatorul trebuie sa aiba posibilitatea de a controla scena prin intermediul mouse-ului si a tastaturii.

Complexitatea scenei a depins foarte mult atat de obiectele texturate pe care le contine, cat si de animatiile care ii confera un caracter interactive.

Folosind librariile OpenGL, GLFW si GLM, am reusit sa realizez urmatoarele aspecte:

* Intrarea in scena cu ajutorul unei animatii de prezentare
* Vizualizarea scenei: scalare, translatie, rotatie, miscarea camerei, atat de la tastatura, cat si cu ajutorul mouse-ului
* Vizualizarea scenei in modurile solid, wireframe, poligonal si smooth
* Utilizarea luminii directionale pentru a lumina toata scena, precum si a unei lumini de tip spotlight, pentru a crea o lumina care vine dintr-un felinar
* Maparea texturilor si definirea materialelor pentru un aspect cat mai placut al obiectelor
* Generarea umbrelor si a unui cub de lumina ce simuleaza miscarea soarelui de-a lungul unei zile
* Fotorealism, prin introducerea efectului de ceata si de ploaie, si, de asemenea, prin animarea diferitelor obiecte(ozn si pisica).

**2. Scenariul**

2.1. Descrierea scenei si a obiectelor

Scena a fost construita cu ajutorul software-ului grafic ”Blender” si cu ajutorul unor modele 3D cu extensia ”.obj” preluate de pe diferite site-uri care propun diverse modele 3D. Aceasta este relativ simpla, continand un numar mediu de obiecte, prezentand organizarea unei gospodarii, care contine o casa, un hambar cu lemne langa el, spatiu de joaca si un observator de lemn pentru copii, incapere cu piscina, banca in curte, o pisica care se misca langa banca, o padure de copaci, un felinar care lumineaza noaptea si un ozn care se roteste deasupra scenei .

O imagine care conține iarbă, cer

Descriere generată automat

2.2. Functionalitati

In cadrul scenei exista mai multe posibilitati de navigare: navigarea prin intermediul tastaturii si navigarea prin intermediul mouse-ului. Au mai fost implementate functionalitati pentru lumini directionale si punctiforme, shadere pentru obiecte, shadere pentru SkyBox, shadere pentru crearea hartii de adancime dupa care vor fi desenate si umbrele. O functionalitate in plus este crearea animatiilor in scena. In scena creata de mine, animatiile sunt reprezentate de un ozn care se roteste in jurul scenei si de o pisica care se misca in fata si in spate, pe axa X. De asemenea, mai exista functionalitati care se modifica odata cu apasarea tastelor, precum pornirea si oprirea unei cete care creste pe masura ce ne departam de obiecte, pornirea si oprirea ploii, modificarea la noapte si inapoi la zi, miscarile umbrelor. Tot prin intermediul tastaturii, scena poate fi vizualizata in mai multe moduri: solid, wireframe si punctiform.



Utilizatorul se poate deplasa prin scena cu ajutorul mouse-ului si a tastelor **W, A, S, D, 8, 9.**

* Tasta W: inaintare in scena.
* Tasta A: deplasare stanga.
* Tasta D: deplasare dreapta.
* Tasta S: departare in scena.
* Tasta 8: deplasare in sus.
* Tasta 9: deplasare in jos

De asemenea, utilizatorul poate roti scena în sus, în jos, la stânga și la dreapta cu ajutorul mouse-ului.

Utilizatorul poate alege sa vizualizeze scena utilizand un alt mod de prezentare. Aceste moduri se pot schimba cu ajutorul tastelor C, X,Z care au urmatorul efect:

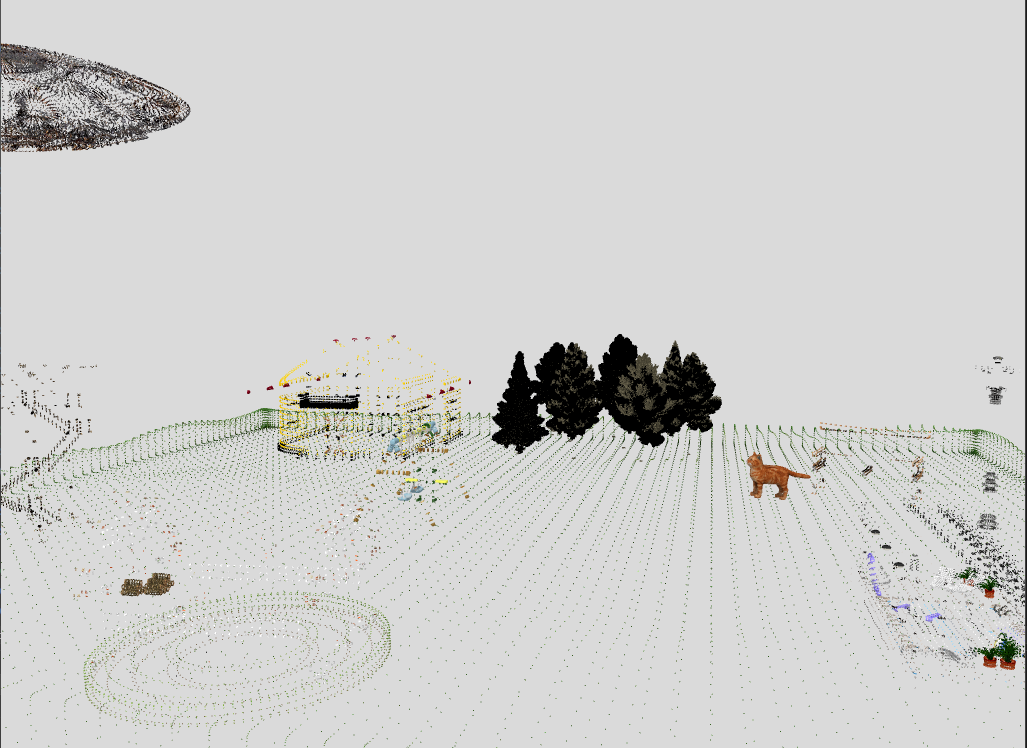
* Tasta C: vizualizarea scenei in modul de reprezentare smooth/ plat.



* Tasta X: vizualizarea scenei in modul de reprezentare wireframe**.**



* Tasta Z: vizualizarea scenei in modul de reprezentare punct**.**



Tot in cadrul scenei, avem efectul de ceata si efectul de ploaie, pe care utilizatorul le poate activa prin tastele G, respectiv P, si le poate opri apasand tasta H.

* Tasta G: activare ceata

A picture containing sky, outdoor, outdoor object, day

Description automatically generated

* Tasta P: activare ploaie



* Tasta H: dezactivare ploaie si ceata

**3. Detalii de implementare**

3.1. Functii si algoritmi

Unul dintre cei mai important pasi ai proiectului este punerea obiectelor in scena, prin operatia de incarcare a modelului 3D. Apoi, modelul mai trece prin 2 etape: render pentru generarea umbrelor si render pentru plasarea obiectelor in scena.

Avem urmatoarele etape in cod:

1. Incarcarea obiectului:

O imagine care conține text

Descriere generată automat

1. Functia de render a obiectului

O imagine care conține text

Descriere generată automat

O imagine care conține text

Descriere generată automat

1. Apelam functia de rendering, prima data ca sa trecem prin shaderul care contine harta de adancime pentru umbre, iar apoi, prin shaderul obisnuit:

O imagine care conține text, interior, portocaliu, închidere

Descriere generată automat

O imagine care conține text

Descriere generată automat

In scena este implementat efectul de ceata, activat din tasta G din cadrul functiei processMovement(), dar si ploaia, activata prin tasta P, tot in cadrul functiei processMovement(). Ambele pot fi oprite apasand tasta H.

O imagine care conține text

Descriere generată automat

Functia de creare a ploii:

Text

Description automatically generated

Acest cod definește o listă de perechi raindrops în care fiecare pereche conține un vector 3D (glm::vec3) reprezentând poziția unei picături de ploaie și un float care reprezinta viteza picăturii de ploaie. Funcția „generateRaindrop” creează o nouă pereche cu o poziție aleatorie și o viteză pentru o picătură de ploaie. Funcția „moveRaindrops” iterează prin fiecare pereche din lista „raindrops”, actualizând poziția picăturii de ploaie scăzând viteza acesteia din coordonata y. Dacă coordonata y a picăturii de ploaie scade sub -2.5 , o nouă picătură de ploaie este generată și adăugată la listă. Lista este curățată prin eliminarea picăturilor de ploaie a căror coordonată y este sub -2.5.

Lumina globala si lumina punctiforma:

Pentru cele doua surse de lumina am ales sa implementez o lumina globala directionala si o lumina de tip punctifroma, care provine de la felinar. Pentru lumina globala, in cadrul fragment shader-ului exista functia ”computeLightComponents()”. Pentru aceasta lumina sunt folosite componentele: ambientala, difuza si speculara, calculate fiecare individual. Acest tip de lumina este calculat tinand cont si de pozitia observatorului(a camerei) care conteaza pentru componenta speculara. Lumina ambientala , nu vine dintr-o anumita directie, este aproximarea luminii globale imprastiate in jurul scenei si reprezinta zona umbrila a obiectului. Lumina difuza este cea imprastiata in toate directiile de o sursa de lumina, iar cea speculara este reflectata direct de suprafata si se refera la cat de asemanatoare este suprafata cu o oglinda.

Pentru lumina punctiforma, in cadrul fragment shader-ului exista functia: ”CalcPointLight()”. Pentru realizarea acesui tip de lumina a fost nevoie de 3 parametrii (constant, linear si quadratic) preluati din main si transmisi functiei din fragment shader si de un vector care sa specifice pozitia luminilor. Cu cat cei 3 parametrii au o valoare mai mica, cu atat intensitatea luminii va creste. Aceste valori pot fi modificate din tastatura si lasa impresia cresterii intensitatii luminii.

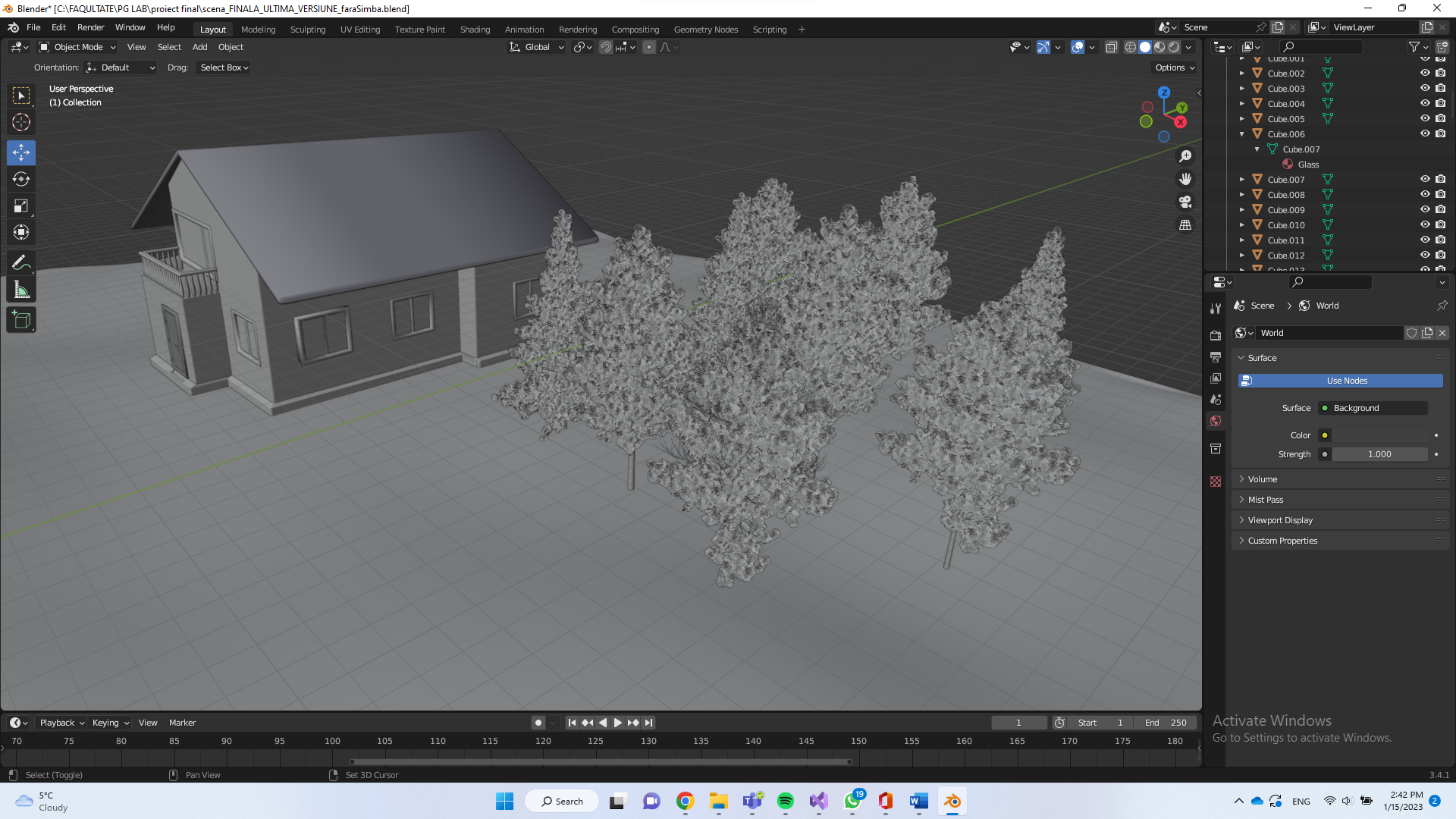
Miscarea camerei :

Pentru explorarea scenei sunt folosite mouse-ul si tastatura. Rotatia camerei se realizeaza cu ajutorul mouse-ului, iar miscarile la stanga sau la dreapta, sau cele de fata, spate cu ajutorul tastaturii. Pentru rotatia camerei cu ajutorul mouse-ului exista in main functia ”mouseCallback” care preia coordonatele initiale ale mouse-ului, iar mai apoi realizeaza rotatia camerei in functie de miscarile mouse-ului, coordonatele pointer-ului acestuia. Pentru navigarea in scena cu ajutorul tastaturii a fost implementata in ”Camera.cpp” functia ”move” care primeste ca si parametru directia de miscare a camerei si viteza cu care aceasta sa se deplaseze in directia indicata de primul parametru.

3.2 Modelul grafic

Modelele grafice sunt realizate pe baza modelelor geometrice care descriu forma si geometria obiectelor, insa cuprind si alte elemente ca textura, culoare, umbre, iluminare, necesare imbunatatirii realismului imaginii obiectului.

Obiectele alese pentru acest proiect au fost preluate de pe internet, urmand ca pozitionarea si texturarea lor sa fie realizate manual, cu ajutorul aplicatiei Blender. Blender nu doar ca permite importul obiectelor si pozitionarea lor relativa, cat si crearea de modele grafice noi.



3.3. Structuri de date

In functie de functionalitatea implementata, am utilizat urmatoarele structuri de date si tipuri speciale de variabile:

- gps::Camera – camera prin care se vizualizeaza scena

- glm::vec3 , mat 4 – pentru transformari geometrice

- gps::Shader – obiect de tip Shader pentru procesare in pipeline

- gps::Model3D – obiect pentru procesarea modelelor 3D

- gps::SkyBox – crearea cubului infinit de jur imprejurul scenei

3.4 Ierarhia de clase

Text

Description automatically generated with low confidence

1. **Prezentarea interfetei grafice utilizator / manual de utilizare**

Controlul camerei se realizeaza cu tastele W,S,A,D,Q,E,8,9 astfel:

* Tasta W: inaintarea in scena
* Tasta A: deplasare stanga
* Tasta D: deplasare dreapta
* Tasta S: departare in scena
* Tasta Q: rotire dreapta
* Tasta E: rotire stanga
* Tasta 8: deplasare in sus
* Tasta 9: deplasare in jos

Utilizatorul poate sa aleaga sa vizualizeze scena in mai multe moduri de prezentare, care se pot schimba cu ajutorul tastelor C,X,Z:

* + Tasta C: vizualizarea scenei in modul de reprezentare smooth
  + Tasta X: vizualizarea scenei in modul de reprezentare wireframe
  + Tasta Z: vizualizarea scenei in modul de reprezentare polygonal

Directia de vizualizare poate fi influentata si de pozitia mouse-ului.

Legenda tastelor pentru activarea anumitor efecte si animatii este prezentata mai jos:

* + Tasta T: animatie de prezentare
  + Tasta R: oprire animatie de prezentare
  + Tasta N: activare spotlight(noapte)- lumini punctiforme
  + Tasta B: lumini directionale
  + Tasta G: activarea efectului de ceata
  + Tasta P: activarea efectului de ploaie
  + Tasta H: oprirea efectului de ceata sau ploaie
  + Tasta L: rotire la stanga a cubului de lumina
  + Tasta J: rotire la dreapta a cubului de lumina
  + Tasta M: harta de adancime
  + Tasta 0: pentru afisarea coordonatelor punctului de pe scena in care ma aflu

1. **Concluzii si dezvoltari ulterioare**

Proiectul a fost un mod bun si creativ de familiarizare cu pricipiile OpenGL, dar si cu softwareul grafic 3D ”Blender” in care s-a dezvoltat scena initial. Opengl prezinta un pipeline grafic destul de puternic pentru dezvoltarea de aplicatii 2D/3D. Ca dezvoltari ulterioare, s-ar putea implementa intrarea in casa, lasarea serii in timp, mai multe animatii.

1. **Bibliografie**

<https://learnopengl.com/>

<https://learnopengl.com/Lighting/Multiple-lights>

<https://free3d.com/>