Canviar *initializeGL* () (treure viewTransform(); i projectTransform();i afegir les funcions:

calculs\_noretall();

initialize\_Camera ();

void MyGLWidget::initialize\_Camera ()

{

viewTransform();

//Calcul dels parametres inicials per a una camara perspectiva

FOV\_inicial = FOV = 2.0\*asin(radiEsc/distancia);//(float)M\_PI/2.f;

ra = 1.f;

//Calcul de parametres inicials per a una camara orthogonal

r = t = radiEsc;

l = b = -radiEsc;

projectTransform();

}

ABAIX -- NOTA: S’HA DE MODIFICAR EN RELACIO AL TERRA I A L’ALTURA MAXIMA

void MyGLWidget::calculs\_noretall() {

//L'alçada del patricio max

alcada\_max = alcadaModel;

//Prenem 5 i -5 pqe son els maxims i els minim del terra (i de l'escena per tant)

maxim\_escena = glm::vec3(5, alcada\_max, 5);

minim\_escena = glm::vec3(-5, 0, -5); //alçada min = -1, que es on es situa el terra

radiEsc = glm::distance(maxim\_escena,minim\_escena) / 2;

distancia = radiEsc\*2.f;

//VRP = (maxim\_escena+minim\_escena); VRP /= 2;

VRP[0] = (maxim\_escena.x+minim\_escena.x)/2.0;

VRP[1] = (maxim\_escena.y+minim\_escena.y)/2.0;

VRP[2] = (maxim\_escena.z+minim\_escena.z)/2.0;

}

MODIFICAR EL projectTransform i viewTransform

void MyGLWidget::projectTransform ()

{

glm::mat4 Proj; // Matriu de projecció

if (perspectiva)

Proj = glm::perspective(FOV, ra, radiEsc, 3.0f\*radiEsc);

else

Proj = glm::ortho(l, r, b, t, radiEsc, 3.0f\*radiEsc);

glUniformMatrix4fv (projLoc, 1, GL\_FALSE, &Proj[0][0]);

}

void MyGLWidget::viewTransform ()

{

glm::mat4 View; // Matriu de posició i orientació

View = glm::translate(glm::mat4(1.f), glm::vec3(0, 0, -distancia)); //Allunyem camara

View = glm::rotate(View, -angleX, glm::vec3(1, 0, 0));

View = glm::rotate(View, -angleY, glm::vec3(0, 1, 0));

View = glm::translate(View, -VRP); // Portem la camara al VRP

glUniformMatrix4fv (viewLoc, 1, GL\_FALSE, &View[0][0]);

}

Afegir angleX i alcadaModel a la constructora (será la maxima y de la nostre escena)

angleY = angleX = 0.0;

alcadaModel = 4;

//(int)((maxz-minz)); sha de fer on el ini camera per que sha de calcular maxz minz

En el void MyGLWidget::*mouseMoveEvent*(QMouseEvent \*e) afegir el calcul d angleX

angleX += (e->y() - yClick) \* M\_PI / 180.0;

I en el .h afegir mirar si ja esta el **angleX**, radiEsc…

void calculs\_noretall();

void initialize\_Camera();

float FOV\_inicial, FOV, ra;

float l, r, b, t;

float distancia, alcada\_max;

int alcadaModel;

glm::vec3 VRP;

glm::vec3 maxim\_escena, minim\_escena;

Modificar la Funcio resizeGL()

void MyGLWidget::*resizeGL* (int w, int h)

{

ra = (float)w/(float)h; // ra(v) = raw

//Recalculem el FOV per tal que no es retalli quan raw<1

if (w < h) {

FOV = 2 \* atan(tan(FOV\_inicial/2.f)/ra);

b = -radiEsc / ra;

t = radiEsc / ra;

}

if (w > h) {

l = -radiEsc \* ra;

r = radiEsc \* ra;

}

glViewport(0, 0, w, h);

projectTransform();

}

**MIRAR QUE ALCADAMODEL ESTIGUI INICIALITZAT A ALGO**