**Camera:**

De maneira a conseguir uma aplicação o mais modular possível decidimos que a manipulação da camara seria feita numa classe à parte da *engine*. Assim sendo criamos uma classe *Camera* que tinha as seguintes funções:

1. Movimento horizontal (esquerda e direita)
2. Movimento vertical (para cima e para baixo)
3. Aproximação/afastamento
4. Velocidade nos movimentos regulável.

Para que este componente conseguisse realizar estas manipulações precisava de guardar os seguintes parâmetros:

* Distancia à origem (raio)
* Ponto com as coordenadas atuais da camara
* Ângulo para a movimentação horizontal (alpha)
* Ângulo para a movimentação vertical (beta)
* Valor que define a variação do angulo (angulo)
* Variáveis que controlassem o movimento do rato(mouseX,mouseY,mouseTracking).

Após definirmos os parâmetros necessários construímos os métodos necessários. Como o tratamento das teclas pressionadas e do movimento do rato são controlados nesta classe precisávamos de métodos que controlavam as teclas e outros métodos que controlavam o rato.

Para as teclas decidimos criar o *normalKeys* e o método *specialKeys.* O primeiro é chamado quando deteta que o sinal mais/menos é pressionado e aumenta/diminui a variável raio para termos um efeito de aproximação/afastamento em relação ao foco (origem). O segundo método é utilizado para controlar as rotações e velocidade dessas mesmas rotações. As setas esquerda e direita são utilizadas para controlar o movimento horizontal (angulo alpha) e consequentemente quando são pressionadas aumentam/diminuem o valor desse angulo. O mecanismo utilizado para controlar o movimento vertical é idêntico, mas neste caso o angulo a alterar é o beta. Além disso utilizamos também as teclas PAGE\_DOWN e PAGE\_UP para definir a velocidade de rotação. Sempre que estas são clicadas a variável que guarda a variação do angulo é multiplicada por um fator que o aumenta/diminui criando um efeito de aceleração/desaceleração.

COLOCAR AQUI A EXPLICAÇÃO DO RATO

Sempre que efetuamos alterações em algum parâmetro que altere a posição da camara recorremos ao método *atualizaPosicao*  que calcula a nova coordenada da camara e atualiza a variável de instancia responsável por guardar essa mesma posição. (posicao).

O método *getPosicao*  é constantemente chamado na render scene da engine e devolve a posição atual da camara após todas as alterações feitas, ou seja, com as teclas/rato estamos constantemente a alterar os parâmetros da camara e consequentemente a sua posição no entanto essa posição só é realmente atualizada quando a imagem é impressa ( só aqui é que é importante verificar a posição atual da camara).