

Computação Grafica e Interfaces

Trabalho Prático 2

Visualizador de Objectos 3D

Autores:

André Correia, 41626

Henrique Garcês, 42077

Prof.: Nuno Correia

Turno: P7

A aplicação desenvolvida permite o carregamento de modelos 3D no formato *obj* através de uma opção do menu. O ficheiro é então passado para a classe *ObjectLoader()* onde é feito o parse e são armazenadas as coordenadas dos vertices, dos vertices da textura e das faces do modelo. Nesta classe também é feito o calculo do factor de escala e do centro do objecto através dos métodos *getScaleFactor()* e *getCenter()*.

Modos de Visualização e Projecções

A visualização dos objectos é feita nos seguintes modos:

- *wireframe* onde são apresentados as arestas dos poligonos que compõem a superficie do objecto;
- *solid fill* onde os poligonos são preenchidos. Este modo é o menos interessante, mas é o que permite aplicar texturas aos modelos;
- *solid/wireframe* que é a combinação dos dois anteriores. Neste modo o modelo é renderizado duas vezes, primeiro com o preenchimento dos poligonos e a seguir no modo wireframe. Apesar de estar disponivel em todos os modos, a aplicação da tecnica de remoção de superficies e arestas ocultas é mais visivel neste modo.

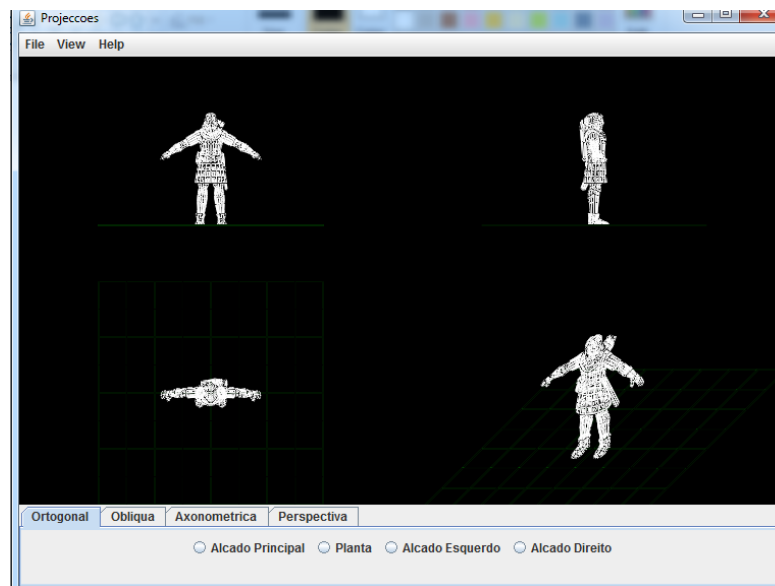
Relativamente às projecções é possivel escolher entre uma única projecção ou projecções multiplas. As projecções multiplas seguem o método europeu onde temos o alçado principal, o alçado lateral esquerdo, a planta e mais uma projecção adicional.

A projecção adicional é escolhida num menu e pode ser uma das seguintes:

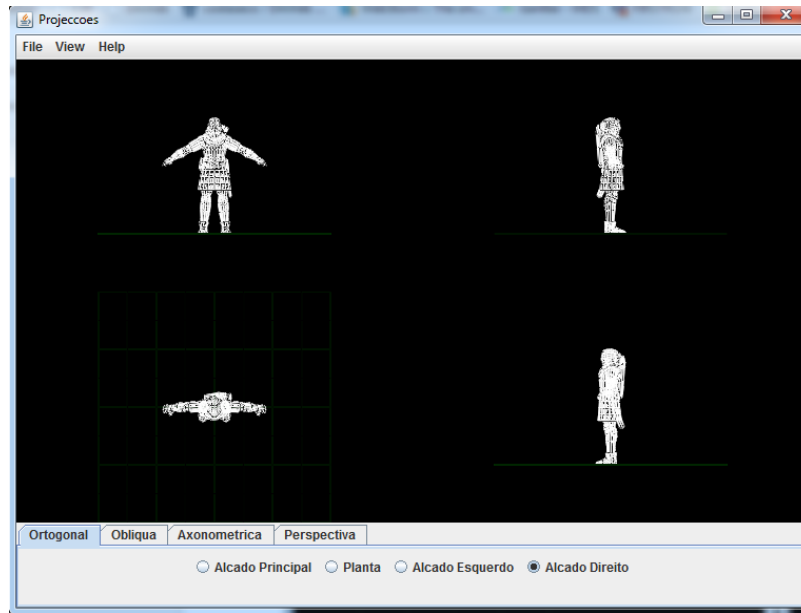
- Projecção Ortogonal: são aplicadas rotações;
- Projecção Obliqua: é aplicada a matriz presente na classe *ObliqueProjection*. Os valores do ângulo de fuga *alpha* e do factor de redução *l* são indicados pelo utilizador através de sliders;
- Projecção Axonometrica: são aplicadas rotações nos eixos dos *xx* ou dos *yy* de acordo com os valores dos ângulos *theta* e *gamma*, os quais o utilizador pode ir alterando através de sliders;
- Projecção Perspectiva: é aplicada a matriz presente na classe *PerspectiveProjection*. O valor do parâmetro *d* pode ser alterado através de um slider. A perspectiva tem projecção no plano $z = 0$ e um ponto de fuga.

A aplicação permite também o carregamento de imagens para serem usadas como textura. Para isto é utilizado a classe *ObjectTexture*, onde a imagem é carregada para um buffer que por sua vez será utilizado no método *loadTexture(GL2 gl)* na função do opengl que especifica uma textura (*glTexImage2D*). A aplicação de uma textura nos poligonos de um modelo só é possível quando o modo de visualização é o *solid fill*.

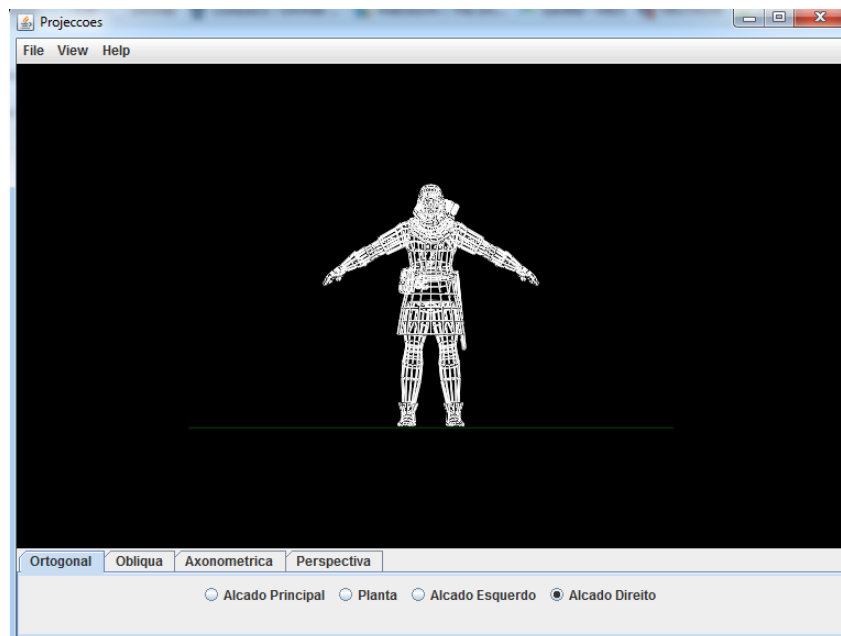
Alguns screenshots com funcionalidades da nossa aplicação:



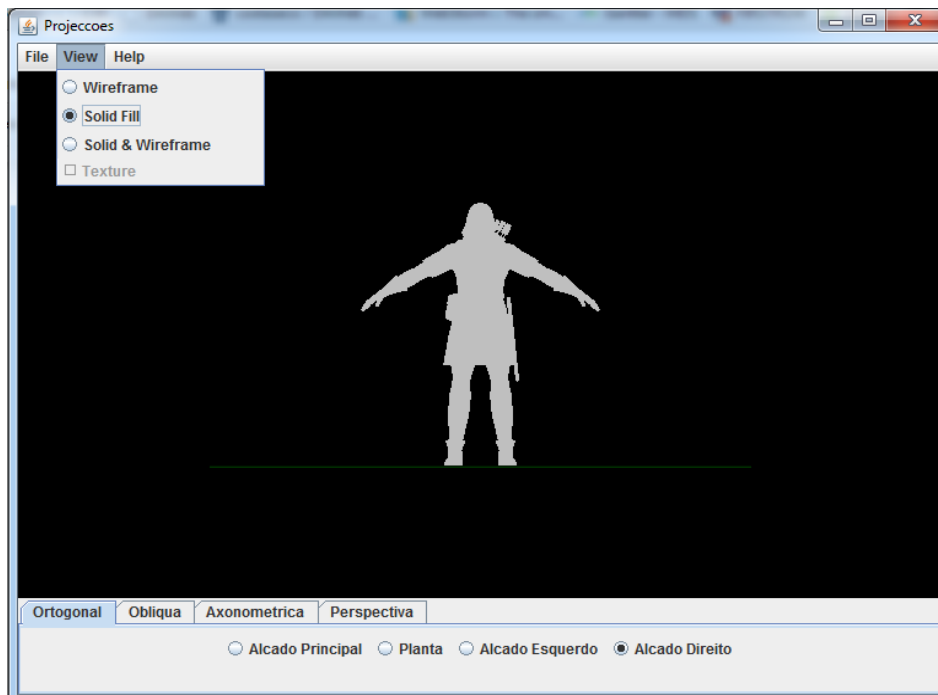
1. Abertura de um objecto com textura por default wireframe. Aqui é possível ver o objecto em várias perspectivas.



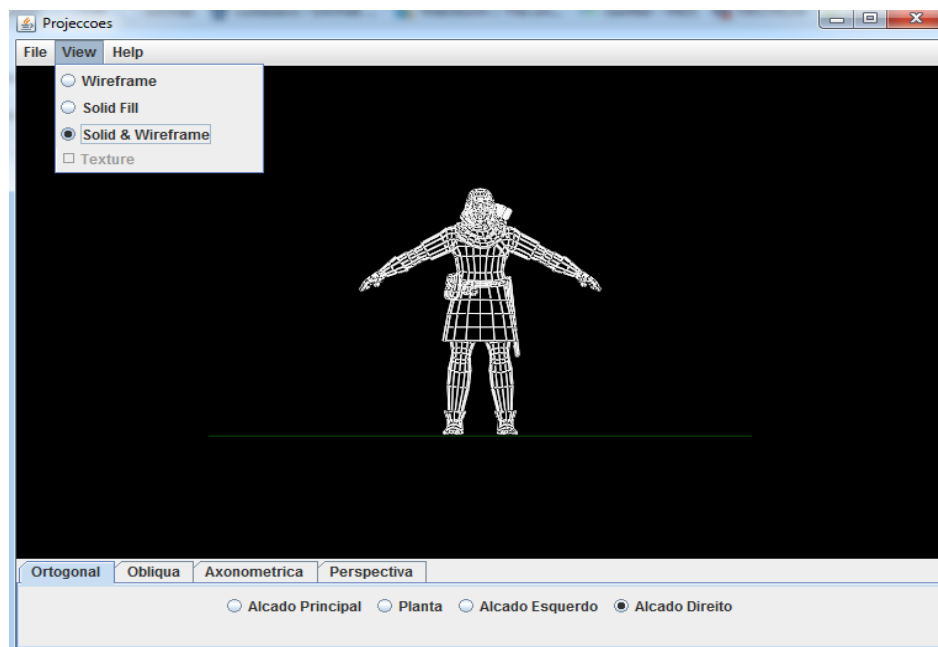
2. Selecção de um RadioButton em Projecção Ortogonal. Poderá escolher quatro perspectivas diferentes: Alçado Principal , Planta, Alçado Esquerdo e Direito. Quando utiliza esta opção a aplicação retorna o resultado apenas no quarto quadrante.



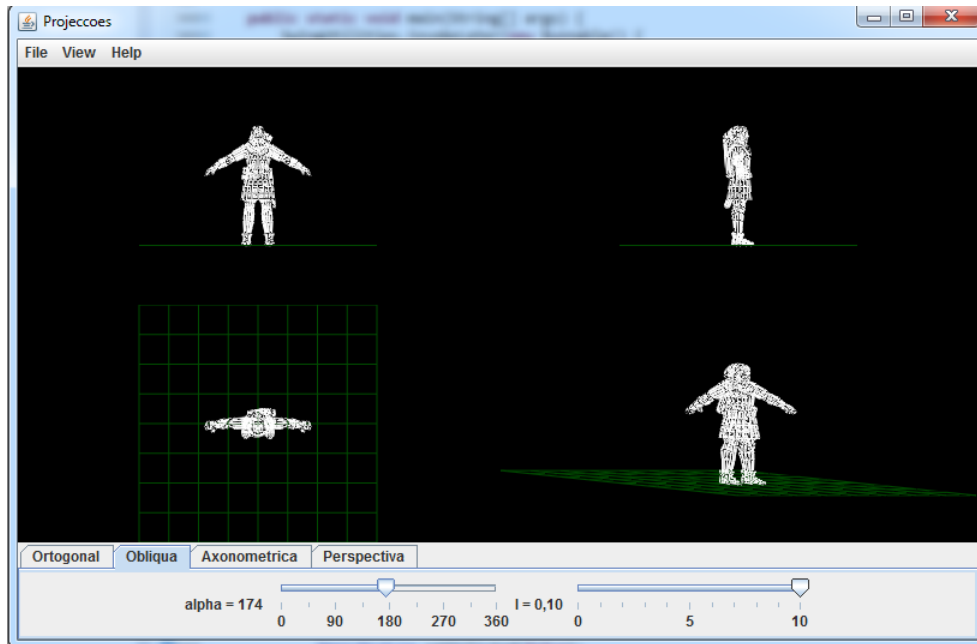
3. Selecção de uma das 4 vistas em grande plano. Para esta opção basta escolher entre os botões 1 a 4



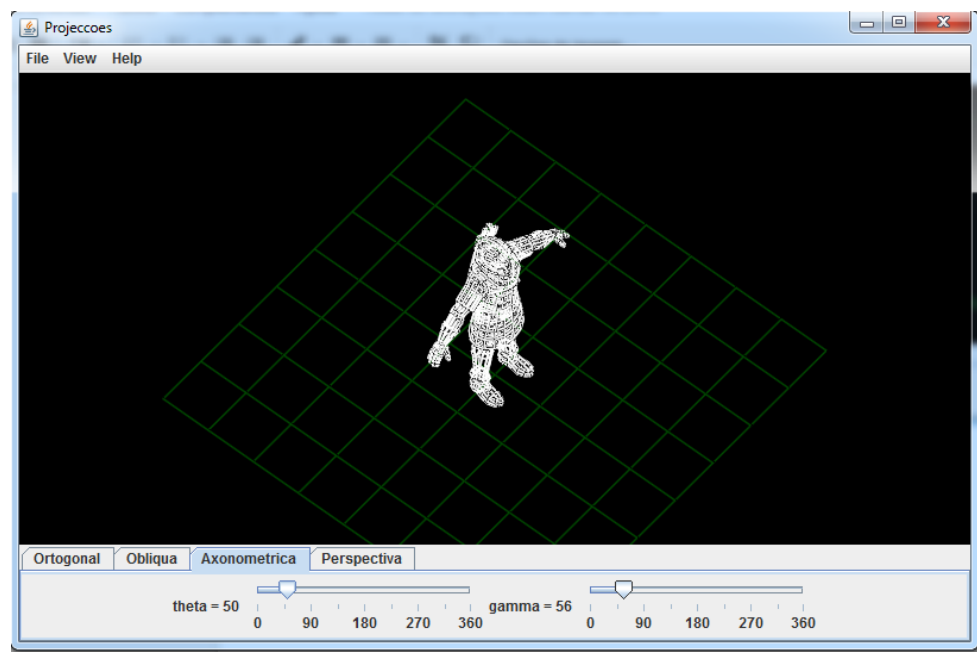
4- Seleção Solid Fill.



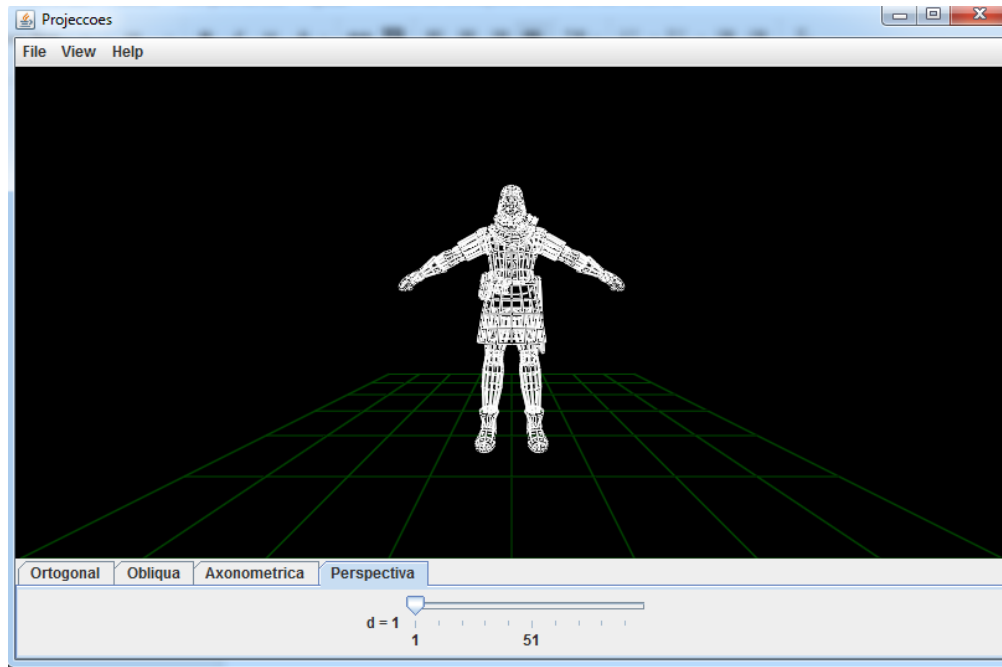
5- Seleção Solid & Wireframe.



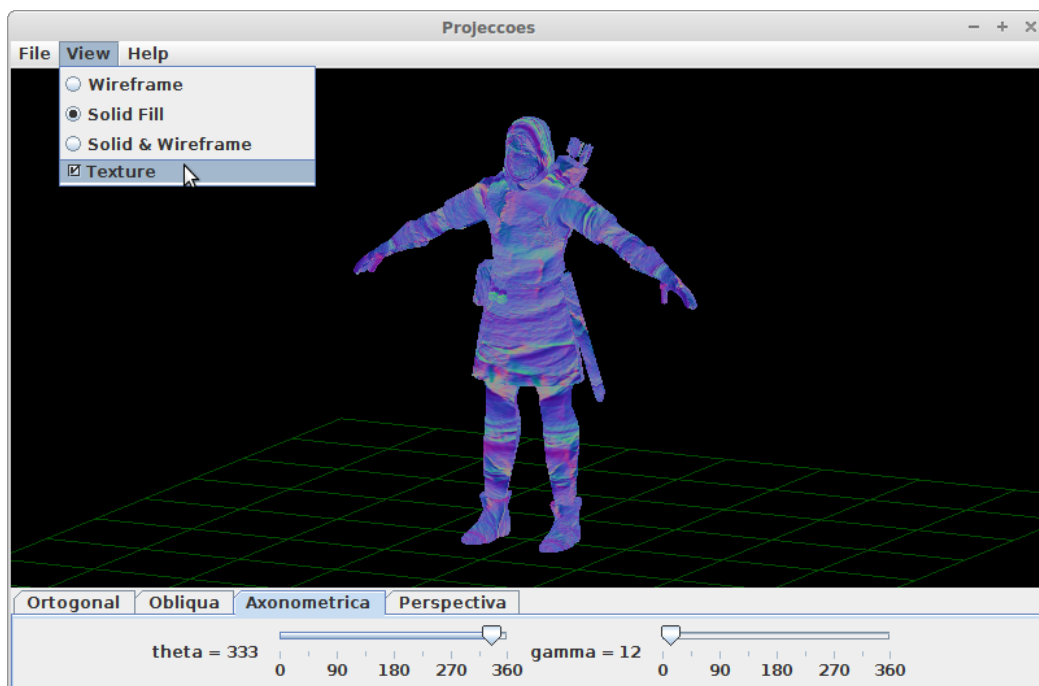
6- Selecção da Tab Obliqua. Esta funcionalidade permite ao utilizador fazer rotações no quarto quadrante através dos Sliders da aplicação. Pode alterar o angulo alpha e também “l”.



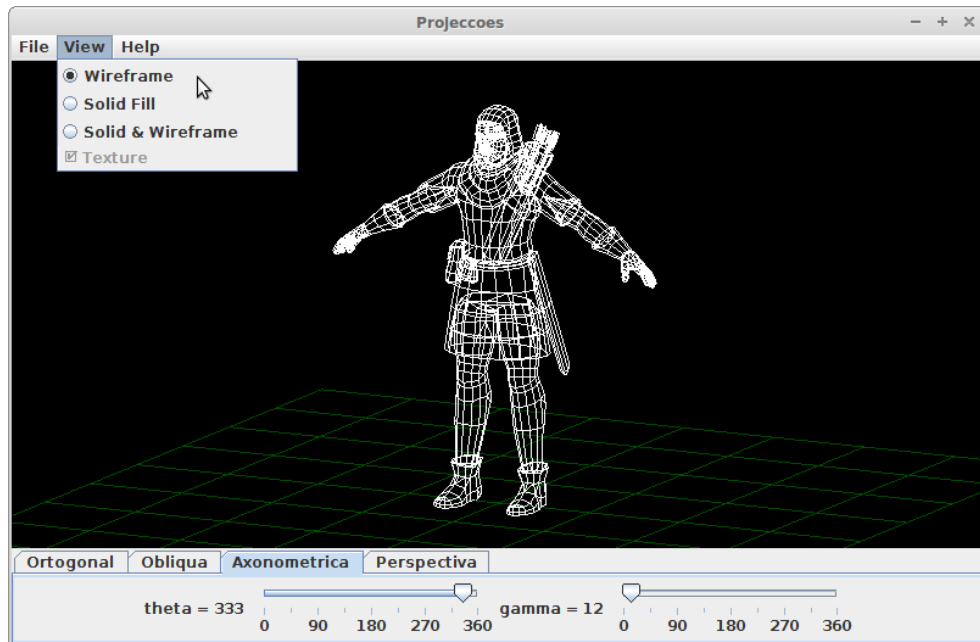
7- Selecção da Tab Axonométrica. É possível fazer rotações dado um ângulo theta e gamma definidos nos Sliders da aplicação.



8- Selecção da Tab Perspectiva. Através dos Sliders é possível visualizar o objecto de várias perspectivas modificando a variável “d”.



9- Aplicação de um textura no modelo.



10- Mesmo que haja uma textura no buffer, a aplicação da mesma só é possível no modo Solid Fill.