

¿Qué ofrece Autentia Real Business Solutions S.L?

Somos su empresa de **Soporte a Desarrollo Informático**.
 Ese apoyo que siempre quiso tener...

1. Desarrollo de componentes y proyectos a medida



2. Auditoría de código y recomendaciones de mejora

3. Arranque de proyectos basados en nuevas tecnologías

1. Definición de frameworks corporativos.
2. Transferencia de conocimiento de nuevas arquitecturas.
3. Soporte al arranque de proyectos.
4. Auditoría preventiva periódica de calidad.
5. Revisión previa a la certificación de proyectos.
6. Extensión de capacidad de equipos de calidad.
7. Identificación de problemas en producción.



4. Cursos de formación (impartidos por desarrolladores en activo)

Spring MVC, JSF-PrimeFaces /RichFaces,
 HTML5, CSS3, JavaScript-jQuery

Gestor portales (Liferay)
 Gestor de contenidos (Alfresco)
 Aplicaciones híbridas

Tareas programadas (Quartz)
 Gestor documental (Alfresco)
 Inversión de control (Spring)

Control de autenticación y
 acceso (Spring Security)
 UDDI
 Web Services
 Rest Services
 Social SSO
 SSO (Cas)

JPA-Hibernate, MyBatis
 Motor de búsqueda empresarial (Solr)
 ETL (Talend)

Dirección de Proyectos Informáticos.
 Metodologías ágiles
 Patrones de diseño
 TDD

BPM (jBPM o Bonita)
 Generación de informes (JasperReport)
 ESB (Open ESB)

Estas en: [Inicio](#) [Tutoriales](#) Detalles del juego de la moto.

Últimas Noticias

- » Historia de la Informática. Capítulo 66. 1988
- » Historia de la informática. Capítulo 65. 1987
- » Autentia en la Sun Open Communities Forum
- » Comentario del libro: El economista naturalista de Robert Frank
- » Contratos ágiles: Vendiendo Scrum a tus clientes.
- » Resumen de la cuarta charla gratuita de Autentia: SCRUM (con video)
- » Si se pregunta ¿Qué ofrece este Web?
- » Vota AdictosAltrabajo.com en DZone
- » Autentia cumple 6 años

+Noticias Destacadas

- » Contratos ágiles: Vendiendo Scrum a tus clientes.
- » Quinta charla Autentia + Projectalis + Agile Spain: Contratos ágiles: Vendiendo Scrum a tus clientes
- » Lo mejor de esta semana: Curso de Scrum con Ángel Medinilla
- » Resumen de la cuarta charla gratuita de Autentia: SCRUM (con video)

+Comentarios Cómico

+Enlaces

Tutorial desarrollado por



Javier Ceballos Fernández

Consultor tecnológico de desarrollo de proyectos informáticos.

Ingeniero en Informática por la Universidad de Alcalá de Henares.

Puedes encontrarme en [Autentia](#)

Somos expertos en Java/J2EE

Catálogo de servicios de Autentia

Descargar (6,2 MB)

Descargar en versión comic (17 MB)

AdictosAlTrabajo.com es el Web de difusión de conocimiento de Autentia.



Catálogo de cursos

Descargar este documento en formato PDF: [DetallesDelJuego.pdf](#)

Fecha de creación del tutorial: **2009-07-15**

Detalles del juego de la moto.

En este tutorial veremos algunos puntos interesantes sobre el tutorial [JMonkeyEngine, Creación de nuestro primer juego.](#)

Nos centraremos en el seguimiento de cámara, los movimientos de la moto y os mencionare algún bug que tiene.

Empecemos por la cámara de seguimiento, todo los cambios para la cámara se harán en el fichero juego.java, para implementar una cámara de seguimiento, tenemos que declarar 2 variables: la primera es la cámara y la siguiente el seguidor.

```
view plain print ?
01. private Camera cam;
02. private ChaseCamera chaser;
```

Ya que se necesita una cámara para poder estar viendo los objetos, luego esta cámara se engloba dentro del seguidor que sera luego quien realmente vaya detrás de la moto.

Luego tendremos que modificar nuestro método update, ya que hay que ir refrescando la posición de la cámara para poder ver la moto en su posición actual. Como hacemos en todos las aplicaciones.

```
view plain print ?
01. skybox.setLocalTranslation(cam.getLocation());
02.
03. if(cam.getLocation().y < (tb.getHeight(cam.getLocation())2)) {
04.     cam.getLocation().y = tb.getHeight(cam.getLocation()) +2;
05.     cam.update();
06. }
```

Después en el método initSystem() tambien tendremos que añadir codigo, ya que es el método de inicio y tendremos que iniciar la cámara que usara el seguidor.

```
view plain print ?
01. try {
02.     display = DisplaySystem.getDisplaySystem (settings.getRenderer());
03.     display.setMinStencilBits(8);
04.     display.createWindow(width, height, depth, freq fullscreen);
05.
06.     cam = display.getRenderer().createCamera(width, height);
07. } catch (JmeException e) {
08.     logger.log(Leve1.SEVERE, "Could not create displaySystem", e);
09.     System.exit(1);
10. }
11.
12. display.getRenderer().setBackgroundColor(ColorRGBA.black.clone());
13.
14. cam.setFrustumPerspective(45.0f, (float) width / (float) height, 1,
15.     5000);
16. cam.setLocation(new Vector3f(200,1000,200));
17.
18. cam.update();
19.
20. timer = Timer.getTimer();
21.
22. display.getRenderer().setCamera(cam);
```

Y por ultimo añadimos un método que llamaremos en InitGame(), a este método le llamaremos buildChaseCamera() en el que definimos las propiedades del seguidor y le asociamos la cámara y el objeto a seguir.

```
view plain print ?
01. private void buildPassManager() {
02.     passManager = new BasicPassManager();
03.
04.     // Add skybox first to make sure it is in the background
05.     RenderPass rPass = new RenderPass();
06.     rPass.add(skybox);
07.     passManager.add(rPass);
08.
09.     shadowPass.add(scene);
10.     shadowPass.addOccluder(player);
11.     shadowPass.setRenderShadows(true);
12.     shadowPass.setLightingMethod(ShadowedRenderPass.LightingMethod.Modulative);
13.     passManager.add(shadowPass);
14. }
```

Ahora os hablare de como hacer los movimientos de la moto, empezare por el movimiento de las ruedas, esto lo haremos dentro del archivo moto.java. Lo que tenemos que hacer son los cálculos matemáticos para calcular el angulo de movimiento, según la velocidad y luego aplicárselo a las ruedas(para hallar los nombres de los nodos de las ruedas y lo que hay que hacer para usarlas mirar el tutorial Scene Monitor, JMonkeyEngine).

La variable wheelAxis es una constante que es del tipo Vector3f cuyo valor es (0,1,0).

Catálogo de servicios Autentia (PDF 6,2MB)



En formato comic...



Web
www.adictosaltrabajo.com

Buscar

Últimos tutoriales

2009-07-15
[Detalles del juego de la moto.](#)

2009-07-14
[JMonkeyEngine, Creación de nuestro primer juego.](#)

2009-07-13
[Ajax tests con Selenium: prototype.js e ICEfaces.](#)

2009-07-08
[AOP con AspectJ y Maven](#)

2009-07-07
[Instalación y configuración de Eclipse Galileo](#)

2009-07-07
[Inicioarse en el manejo de JME, Creación de un Cloth.](#)

2009-07-06
[Primeros pasos con Blender: Pintando nuestra mascota en 3D](#)

2009-07-06
[DBUnit-Exportar e Importar BBDD](#)

2009-07-05
[JMeter, Pruebas de stress sobre aplicaciones web: Grabando y reproduciendo navegaciones](#)

2009-07-02
[Axis2: Invocación de Servicios Web usando distintos MEP](#)

2009-07-02
[Instalación OpenOffice](#)

2009-07-02
[Juegos 3D en Java: Blender y JMonkeyEngine](#)

2009-06-20
[StAX \(Xml Pull Parser\): Streaming API para XML](#)

2009-06-15
[Configuración de la desconexión de usuarios con ICEfaces](#)

2009-06-10
[LWUIT: Una librería gráfica tipo AWT o Swing para J2ME](#)

2009-06-10
[Mapas mentales con XMind](#)

2009-02-26
[Redimensionar Imagenes en Windows Vista](#)

2009-06-08
[UploadFile con Icefaces + Hibernate + Anotaciones](#)

2009-06-05
[Habilitar exportación en Liferay](#)

2009-06-01
[Registrar Liferay en Eclipse](#)

2009-05-29
[Liferay Social Office](#)

2009-05-28
[Broadcast con Ustream](#)

2009-05-25
[Tabla datos accesible con ordenación y paginación](#)

2009-05-21
[Primeros pasos con Audacity: Un editor de sonido libre y multiplataforma.](#)

2009-05-11
[Introducción a TortoiseSVN](#)

```

view plain print ?
01. private void rotateWheels(float time) {
02.
03.     if (vehicleIsMoving()) {
04.         if(velocity > FastMath.FLT_EPSILON) {
05.             angle = angle - ((time) * velocity *0.5f);
06.             if (angle < -360) {
07.                 angle = 0;
08.             }
09.         } else {
10.             angle = angle + ((time) * velocity *0.5f);
11.             if (angle > 360) {
12.                 angle = 0;
13.             }
14.         }
15.         rotQuat.fromAngleAxis(angle, wheelAxis);
16.         frontwheel.getLocalRotation().multLocal(rotQuat);
17.         backwheel.setLocalRotation(frontwheel.getLocalRotation());
18.     }
19. }

```

Para hacer que la moto se tumbe añadimos el método processLean que es el que nos permite hacer que la moto se tumbe cuando gire, entonces para ello tenemos que hacer un calculo del angulo que necesita y lo hacemos en función del tiempo ya que la moto se encuentra en movimiento y según como vaya de rápido inclinara mas o menos.

La variable q es de tipo Quaternion y se usa de variable auxiliar.

```

view plain print ?
01. private void processLean(float time) {
02.     //check if we are Leaning at all
03.     if(lean != 0) {
04.         if(lean == -1 && leanAngle < 0) {
05.             leanAngle += -lean *4 * time;
06.         } else if(lean == 1 && leanAngle > 0) {
07.             leanAngle += -lean *4 * time;
08.         } else {
09.             leanAngle += -lean *2 * time;
10.         }
11.         //max Lean is 1 and -1
12.         if(leanAngle > 1) {
13.             leanAngle = 1;
14.         } else if(leanAngle < -1) {
15.             leanAngle = -1;
16.         }
17.     } else { //we are not Leaning, so right ourself back up.
18.         if(leanAngle < LEAN_BUFFER && leanAngle > -LEAN_BUFFER) {
19.             leanAngle = 0;
20.         }
21.         else if(leanAngle < -FastMath.FLT_EPSILON) {
22.             leanAngle += time *4;
23.         } else if(leanAngle > FastMath.FLT_EPSILON) {
24.             leanAngle -= time *4;
25.         } else {
26.             leanAngle = 0;
27.         }
28.     }
29.
30.     q.fromAngleAxis(leanAngle, leanAxis);
31.     model.setLocalRotation(q);
32.
33.     lean = 0;
34. }

```

Para hacer el efecto de freno sin mas con que disminuyamos el valor de la velocidad en función del tiempo se consigue.

```

view plain print ?
01. public void brake(float time) {
02.     velocity -= time * braking;
03.     if(velocity < -minSpeed) {
04.         velocity = -minSpeed;
05.     }
06. }

```

El efecto de acelerar es el mismo que el de frenar solo que en vez de restar sumamos.

```

view plain print ?
01. public void accelerate(float time) {
02.     velocity += time * acceleration;
03.     if(velocity > maxSpeed) {
04.         velocity = maxSpeed;
05.     }
06. }

```

Para hacer que las ruedas patinen usaremos el método drift

```

view plain print ?
01. public void drift(float time) {
02.     if(velocity < -FastMath.FLT_EPSILON) {
03.         velocity += ((weight.5) * time);
04.         if(velocity > 0) {
05.             velocity = 0;
06.         }
07.     } else if(velocity > FastMath.FLT_EPSILON){
08.         velocity -= ((weight.5) * time);
09.         if(velocity < 0) {
10.             velocity = 0;
11.         }
12.     }
13. }

```

Para que la moto gire hemos implementado una nueva clase que es una extension de KeyInputAction ya que luego lo usaremos para que cuando pulsemos unas teclas especificas ocurran los giros, ha este fichero lo hemos denominado RotacionMoto.java.

2009-05-07
Hacer 'scp' de varios ficheros sin solicitud de clave

2009-05-02
Plugin Hibernate3 para Maven

2009-04-26
AgileDraw: una técnica rápida de modelado

2009-04-24
Spring AOP: Cacheando aplicaciones usando anotaciones y aspectos con AspectJ

2009-04-20
Modelos de conocimiento con CmapTools

2009-04-16
Informes Crosstab con iReport

2009-04-16
Registro de un fichero de datos personales con el formulario NOTA

2009-04-15
Estadísticas de www.adictosaltrabajo.com Abril 2009

2009-04-15
Iniciación a OSWorkflow con Spring

2009-04-14
Tests de Selenium con librerías de componentes JSF: Apache Tomahawk.

2009-04-13
JTAPI. El API de Telefonía para Java

2009-04-13
Registro de Web Services con Apache JUDDI. Configuración y ejemplo

2009-04-13
Cómo hacer UML con Eclipse y el plugin UML2

2009-04-09
Spring WS: Servicios Web a través del correo electrónico

2009-04-02
Creación de cursos con Moodle

2009-03-31
Integrar Liferay Portal 5.2.1 con Pentaho BI 2.0.0 sobre MySQL 5.1

2009-03-31
Spring WS: Construcción de Clientes de Servicios Web con Spring

2009-03-30
Administración de sitios Moodle

2009-03-29
Empaquetamiento de aplicaciones de escritorio (standalone) con Maven

2009-03-27
Primeros pasos con Moodle

2009-03-26
Introducción a JSF Java

2009-03-25
A1 Website Analyzer

2009-03-24
Cómo ver el correo de Gmail sin conexión a Internet

2009-03-20
JasperReports Maven Plugin

2009-03-16
Creación de contenidos SCORM: eXe

Últimas ofertas de empleo

2009-06-29
Atención a cliente - Call Center - BARCELONA.

2009-06-25
Atención a cliente - Call Center - BARCELONA.

2009-06-20
Comercial - Ventas - CASTELLON.

2009-06-19
Otras - Ingeniería (minas, puentes y puertos) - VALENCIA.

2009-06-17
Comercial - Ventas - ALICANTE.

Anuncios Google

```

view plain print ?
01. package Intento1;
02.
03. import com.jme.input.action.InputActionEvent;
04. import com.jme.input.action.KeyInputAction;
05. import com.jme.math.FastMath;
06. import com.jme.math.Matrix3f;
07. import com.jme.math.Vector3f;
08.
09.
10. public class RotacionMoto extends KeyInputAction {
11.     public static final int RIGHT = 0;
12.     public static final int LEFT = 1;
13.     private static final Matrix3f incr = new Matrix3f();
14.     private static final Matrix3f tempMa = new Matrix3f();
15.     private static final Matrix3f tempMb = new Matrix3f();
16.
17.
18.     private Vector3f upAxis = new Vector3f(0,1,0);
19.
20.     private moto vehicle;
21.     private int direction;
22.     private int modifier = 1;
23.
24.
25.     public RotacionMoto(moto vehicle, int direction) {
26.         this.vehicle = vehicle;
27.         this.direction = direction;
28.     }
29.
30.
31.     public void performAction(InputActionEvent evt) {
32.         if(vehicle.getVelocity() > -FastMath.FLT_EPSILON && vehicle.getVelocity() < FastMath.FLT_EPSILON) {
33.             return;
34.         }
35.
36.         if(direction == LEFT) {
37.             modifier = 1;
38.         } else if(direction == RIGHT) {
39.             modifier = -1;
40.         }
41.
42.         if(vehicle.getVelocity() < 0) {
43.             incr.fromAngleNormalAxis(-modifier * vehicle.getTurnSpeed() * evt.getTime(), upAxis);
44.         } else {
45.             incr.fromAngleNormalAxis(modifier * vehicle.getTurnSpeed() * evt.getTime(), upAxis);
46.         }
47.         vehicle.getLocalRotation().fromRotationMatrix(
48.             incr.mult(vehicle.getLocalRotation().toRotationMatrix(tempMa),
49.                 tempMb));
50.         vehicle.getLocalRotation().normalize();
51.         vehicle.setRotateOn(modifier);
52.     }
53. }

```

Para hacer que vaya hacia delante y hacia atrás también tenemos que denominar una nueva clase para esto suceda con una teclas definidas, también la definiremos como una clase extendida de KeyInputAction como la clase anterior y la llamaremos AvanzarRetroceder.java

```

view plain print ?
01. package Intento1;
02.
03. import com.jme.input.action.InputActionEvent;
04. import com.jme.input.action.KeyInputAction;
05.
06. public class AvanzarRetroceder extends KeyInputAction {
07.     public static final int FORWARD = 0;
08.     public static final int BACKWARD = 1;
09.
10.     private moto node;
11.     private int direction;
12.
13.
14.     public AvanzarRetroceder(moto node, int direction) {
15.         this.node = node;
16.         this.direction = direction;
17.     }
18.
19.     public void performAction(InputActionEvent evt) {
20.         if(direction == FORWARD) {
21.             node.accelerate(evt.getTime());
22.         } else if(direction == BACKWARD){
23.             node.brake(evt.getTime());
24.         }
25.     }
26. }

```

Para hacer que derrape hay que hacer exactamente lo mismo que en las dos clases anteriores, definirla como clase extendida de KeyInputAction en este caso lo llamaremos AccionDerrape.java

```

view plain print ?
01. package Intento1;
02.
03. import com.jme.input.action.InputActionEvent;
04. import com.jme.input.action.KeyInputAction;
05.
06. public class AccionDerrape extends KeyInputAction {
07.
08.     private moto vehicle;
09.
10.     public AccionDerrape(moto vehicle) {
11.         this.vehicle = vehicle;
12.     }
13.
14.     public void performAction(InputActionEvent evt) {
15.         vehicle.drift(evt.getTime());
16.     }
17. }
18.
19. }

```

Como bugs que tiene el juego, podríamos decir que es uno que la moto se puede salir del escenario y parece que va volando, eso es porque no están implementadas las colisiones

Para cualquier duda de los métodos y las clases de JMonkey Engine se puede consultar <http://www.jmonkeyengine.com/doc/> :

Espero que les haya sido útil para poder profundizar algo más en el juego.

Esperemos que les haya sido útil este tutorial, seguiremos haciendo más tutoriales sobre esta tecnología analizando más ejemplos algo más complicados, todo el que quiera hacer una aportación será bien recibida.

Para comunicarme cualquier problema o sugerencia de mejora podéis utilizar la zona de comentarios, de este modo todo el mundo se podrá aprovechar de las respuestas.

Saludos.

¿Qué te ha parecido el tutorial? Déjanos saber tu opinión y ivota!

Muy malo Malo Regular Bueno Muy bueno



Animáte y coméntanos lo que pienses sobre este tutorial

Puedes opinar o comentar cualquier sugerencia que quieras comunicarnos sobre este tutorial; con tu ayuda, podemos ofrecerte un mejor servicio.

Nombre: E-Mail:

Comentario:

Enviar comentario

[Texto Legal y condiciones de uso](#)

- Puedes inscribirte en nuestro servicio de notificaciones [haciendo clic aquí](#).
- Puedes firmar en nuestro libro de visitas [haciendo clic aquí](#).
- Puedes asociarte al grupo AdictosAlTrabajo en XING [haciendo clic aquí](#).
- Añadir a favoritos Technorati.  ADD THIS BLOG TO MY FAVORITES

 Esta obra está licenciada bajo [licencia Creative Commons de Reconocimiento-No comercial-Sin obras derivadas 2.5](#)

Recuerda

Autentia te regala la mayoría del conocimiento aquí compartido ([Ver todos los tutoriales](#)). Somos expertos en: J2EE, Struts, JSF, C++, OOP, UML, UP, Patrones de diseño ... y muchas otras cosas.

¿Nos vas a tener en cuenta cuando necesites consultoría o formación en tu empresa?, ¿Vas a ser tan generoso con nosotros como lo tratamos de ser con vosotros?

Somos pocos, somos buenos, estamos motivados y nos gusta lo que hacemos ...

Autentia = Soporte a Desarrollo & Formación.

info@autentia.com



Tutoriales recomendados

Nombre	Resumen	Fecha	Visitas	Valoración	Votos	Pdf
JMonkeyEngine, Creación de nuestro primer juego.	Intentaremos enseñaros a crear vuestro primer juego, partiremos de un ejemplo hecho de JMonkeyEngine, que trata sobre el manejo de una moto por un escenario.	2009-07-14	128	Bueno	9	
Ajax tests con Selenium: prototype.js e ICEFaces.	En este tutorial se va habla de cómo escribir tests funcionales con Selenium IDE sobre aplicaciones que realizan recargas controladas de la interfaz de usuario con Ajax.	2009-07-13	123	Muy bueno	1	
AOP con AspectJ y Maven	Programacion orientada a aspectos con AspectJ y Maven	2009-07-08	265	Bueno	1	
Iniciarse en el manejo de JME, Creación de un Cloth.	Primeros pasos con Jmonkey engine, crearemos una pequeña aplicación animada basada en un Cloth con colisiones	2009-07-07	351	Muy bueno	6	
Primeros pasos con Blender: Pintando nuestra mascota en 3D	Inicio del manejo de blender, os mostraremos paso a paso como se dibuja en blender	2009-07-06	325	Bueno	6	
Juegos 3D en Java: Blender y JMonkeyEngine	En este tutorial se abordara una pequeña iniciación a los juegos 3D en java usando la herramienta Blender y como motor gráfico JMonkeyEngine	2009-07-02	841	Bueno	15	
Creación de cursos con Moodle	En este tercer tutorial de la serie veremos en detalle la creación de un curso	2009-04-02	2959	Muy bueno	11	
Cómo ver el correo de Gmail sin conexión a Internet	En este tutorial vamos a ver como podemos configurar el navegador Firefox 3 para poder acceder a todo nuestro histórico de correos sin necesitar de un conexión a Internet.	2009-03-24	2555	Bueno	12	
Redimensionar Imagenes en Windows Vista	Instalación y uso de ImageResizer	2009-02-26	773	Muy bueno	2	
Introducción teórica a XPath	Con este tutorial se pretende enseñar de forma teórica el uso de expresiones XPath pensando sobre todo en su posible uso con el paquete Selenium u otras aplicaciones.	2009-02-26	1639	Muy bueno	4	

Nota:

Los tutoriales mostrados en este Web tienen como objetivo la difusión del conocimiento. Los contenidos y comentarios de los tutoriales son responsabilidad de sus respectivos autores. En algún caso se puede hacer referencia a marcas o nombres cuya propiedad y derechos es de sus respectivos dueños. Si algún afectado desea que incorporemos alguna reseña específica, no tiene más que solicitarlo. Si alguien encuentra algún problema con la información publicada en este Web, rogamos que informe al administrador rcanales@adictosaltrabajo.com para su resolución.