

**Ingeniería técnica en informática de sistemas**

**Proyecto fin de carrera - Desarrollo de aplicaciones  
para dispositivos móviles (Android)**

**Desarrollo de videojuegos para Android con libGDX  
(Casual games)**

**<http://projects.3comet.com/>**

**Consultores: Marc Domingo Prieto  
Jordi Almirall López**

**Autor: Xavier Figuera Alberich  
xfiguera@uoc.edu**



*This document, Fling the sheep, casual game for Andorid with libGDX  
(c) 2013 by Xavier Figuera Alberich*

*This document, Fling the sheep, casual game for Andorid with libGDX is licensed under a  
Creative Commons Attribution-NonCommercial-ShareAlike 4.0 International License.*



**3** Comet  
Studio

# Índice de contenidos

3 Comet Studio



Diapositiva 4 – Justificación del proyecto	Diapositiva 13 – Diseño “Concept Art”
Diapositiva 5 – Objetivos del proyecto	Diapositiva 14 – Diseño “Concept Art”
Diapositiva 6 – Planificación del proyecto	Diapositiva 15 – Implementación
Diapositiva 7 – Herramientas y tecnologías	Diapositiva 16 – Estructura videojuego
Diapositiva 8 – Riesgos del proyecto	Diapositiva 17 – Entorno de desarrollo
Diapositiva 9 – Análisis funcional	Diapositiva 18 – Patrones de diseño
Diapositiva 10 – Estudio comparativo	Diapositiva 19 – Conclusiones
Diapositiva 11 – Perfiles de usuario	Diapositiva 20 – Lineas abiertas
Diapositiva 12 – Diseño “Concept Art”	Diapositiva 21 – Videojuego en un LG-610v



# Justificación del proyecto

3 Comet Studio



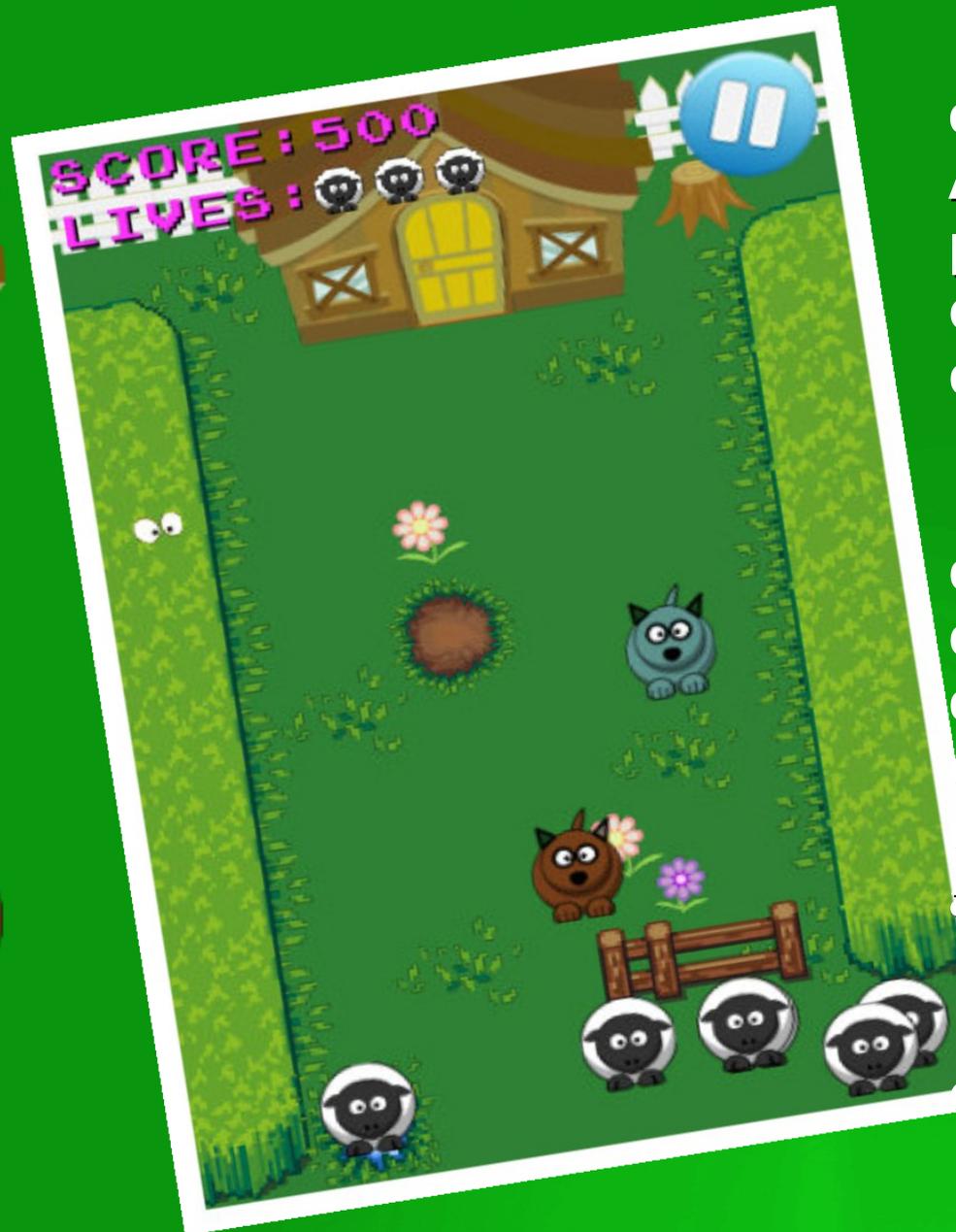
Este proyecto de fin de carrera, se centra en el desarrollo de aplicaciones para dispositivos móviles con sistema operativo Android y concretamente se quiere centrar en un subconjunto de aplicaciones focalizadas al entretenimiento conocidas por todo el mundo como videojuegos.

En estos últimos años los smartphones y tabletas se han convertido en las nuevas plataformas móviles para juegos que compiten con los sistemas portátiles clásicos que generalmente tienden a ser más cerrados.

Existen distintos tipos de géneros de videojuegos, no obstante, este proyecto se centra en el desarrollo de un videojuego tipo casual. Un casual game, se distingue del resto de videojuegos por estar focalizado a grandes masas de público debido a unas reglas sencillas de juego. Estos no requieren de habilidades especiales para poderlos jugar, ni de una dedicación de tiempo elevada. Normalmente, acostumbran a tener unos costes de producción y de distribución bajos comparado con otro tipo de juegos.



# FLING THE SHEEP 3 Comet Studio



## Objetivos del proyecto

El objetivo principal de este proyecto es adquirir experiencia en el desarrollo de videojuegos para Android. Sin embargo, yendo más allá de la plataforma, también existe un interés por ampliar conocimientos en cuanto a programación y diseño de videojuegos en general.

El juego consistirá en conducir unas ovejas descarriadas a su corral, éstas con nuestra ayuda deberán esquivar los acechos de las profundidades del bosque, unos lobos malvados quieren comérselas durante el camino hacia el deseado corral, estos lobos serán el principal peligro que las amenaza, ayudalas !!!

Juego completamente en 2D con un "Look & Feel" cartoon en clave de humor.

**F L I N G**

**T H E**

**S H E E P**



Nombre de tarea	Duración	Comienzo	Fin
<b>Análisis previo planificación</b>	<b>9 días</b>	<b>lun 23/09/13</b>	<b>mar 01/10/13</b>
Diseño conceptual del videojuego	1 día	lun 23/09/13	lun 23/09/13
Definición de las tecnologías	2 días	mar 24/09/13	mié 25/09/13
Análisis de riesgos	2 días	jue 26/09/13	vie 27/09/13
Planificación del proyecto	1 día	sáb 28/09/13	sáb 28/09/13
Redacción del plan de trabajo PEC1	2 días	dom 29/09/13	mar 01/10/13
Entrega PEC1 (Hito)	1 día	mar 01/10/13	mar 01/10/13
<b>Analisis de requisitos</b>	<b>5 días</b>	<b>mié 02/10/13</b>	<b>dom 06/10/13</b>
Definición de casos de uso	2 días	mié 02/10/13	jue 03/10/13
Definición de maquinas de estado	2 días	vie 04/10/13	sáb 05/10/13
Diseño básico estructura aplicación	1 día	dom 06/10/13	dom 06/10/13
<b>Diseño</b>	<b>23 días</b>	<b>lun 07/10/13</b>	<b>mar 29/10/13</b>
Definición de usuarios y contextos de uso del videojuego	2 días	lun 07/10/13	mar 08/10/13
Diseño conceptual	2 días	mié 09/10/13	jue 10/10/13
Selección de arte	4 días	vie 11/10/13	lun 14/10/13
Selección de fuentes	2 días	mar 15/10/13	mié 16/10/13
Selección de personajes y enemigos	2 días	jue 17/10/13	vie 18/10/13
Diseño de pantallas que intervienen (prototipo)	8 días	sáb 19/10/13	sáb 26/10/13
Redacción del documento PEC2	3 días	sáb 26/10/13	mar 29/10/13
Entrega PEC2 (Hito)	1 día	mar 29/10/13	mar 29/10/13
<b>Implementación y pruebas</b>	<b>42 días</b>	<b>mié 30/10/13</b>	<b>mar 10/12/13</b>
Implementación de navegables y contenidos	15 días	mié 30/10/13	mié 13/11/13
Implementación gameplay	15 días	jue 14/11/13	jue 28/11/13
Implementación de sonido y efectos	4 días	vie 29/11/13	lun 02/12/13
Pruebas y refinamiento de la implementación	5 días	mar 03/12/13	sáb 07/12/13
Preparación del APK y redacción de la PEC3	3 días	sáb 07/12/13	mar 10/12/13
Entrega PEC3 (Hito)	1 día	mar 10/12/13	mar 10/12/13
<b>Finalización</b>	<b>29 días</b>	<b>mié 11/12/13</b>	<b>mié 08/01/14</b>
Redacción memoria final proyecto	12 días	mié 11/12/13	dom 22/12/13
Montaje presentación diapositivas	2 días	vie 20/12/13	sáb 21/12/13
Montaje del video final presentación proyecto	10 días	dom 22/12/13	mié 01/01/14
Revisión y entrega final (Hito)	7 días	jue 02/01/14	mié 08/01/14
<b>Total temporización proyecto</b>	<b>108 días</b>	<b>lun 23/09/13</b>	<b>mié 08/01/14</b>

**3 Comet Studio**

**Planificación Proyecto**

# Herramientas y tecnologías

# 3 Comet Studio



## Planificación



## Diseño



INKSCAPE



## Programación

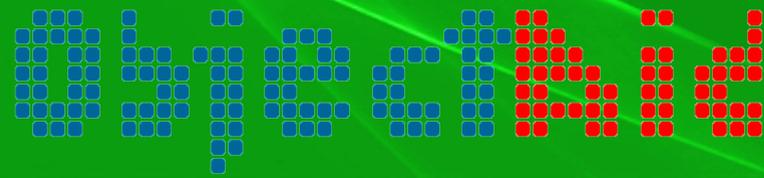


# git



# Java

# libGDX



## Documentación



Apache  
**OpenOffice**™

## Edición de audio



# Audacity



## Edición de video



# Riesgos del proyecto

## 3 Comet Studio



No se dispone de experiencia con la librería escogida ni tampoco con programación sobre Android. A nivel técnico no se prevé que puedan surgir muchos riesgos en líneas generales. No obstante, existe la posibilidad de gastar más tiempo del previsto consultando documentación para lograr los objetivos establecidos.

Existe un riesgo más alto a nivel artístico, ya que no se dispone de conocimientos de arte en profundidad.

En cuanto a la parte de audio la situación es exactamente la misma, no se dispone de ningún conocimiento para componer música ni efectos.

Por lo tanto, se buscarán tanto gráficos como sonidos con licencias libres, intentando que estos queden integrados de la mejor manera posible al videojuego.

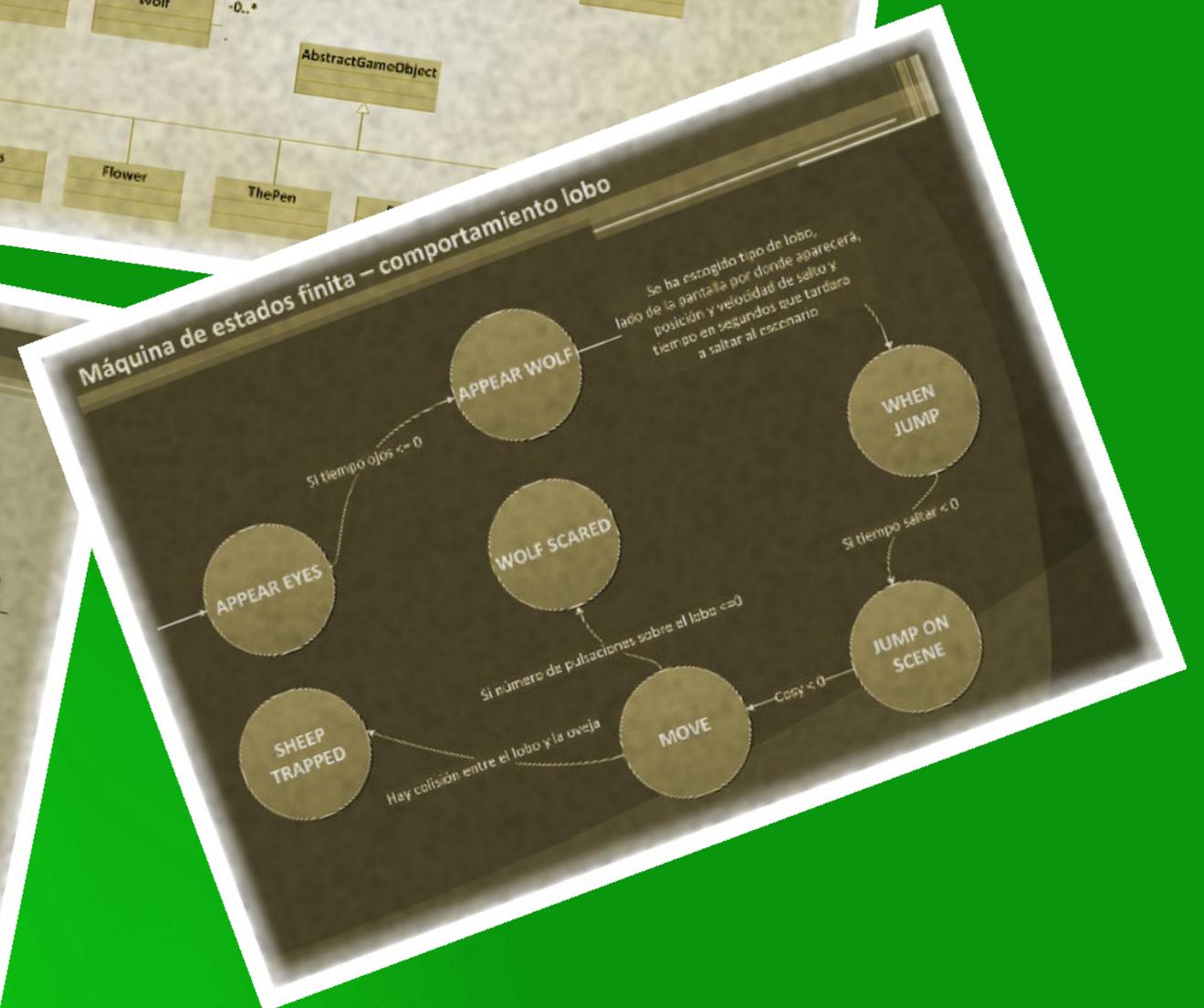
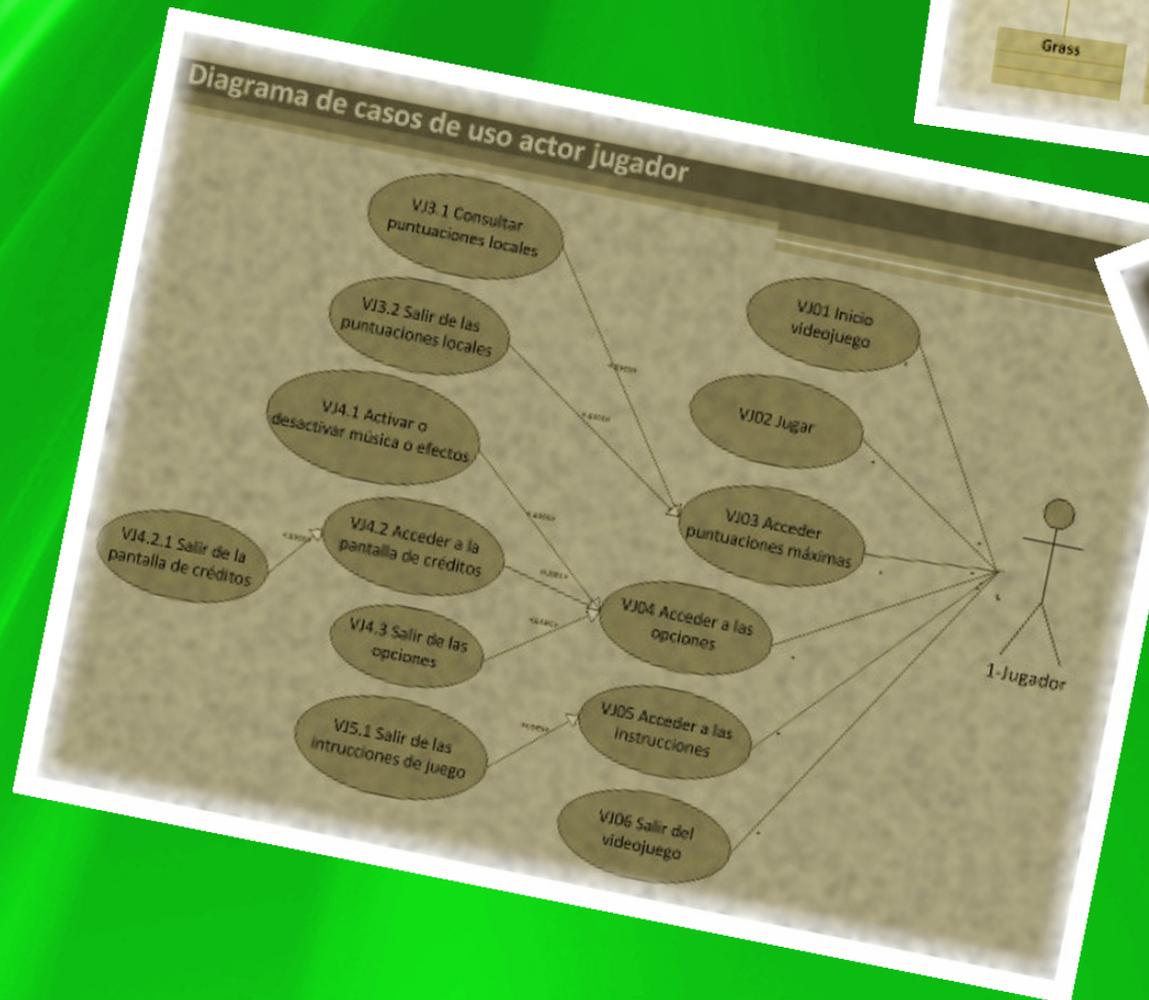
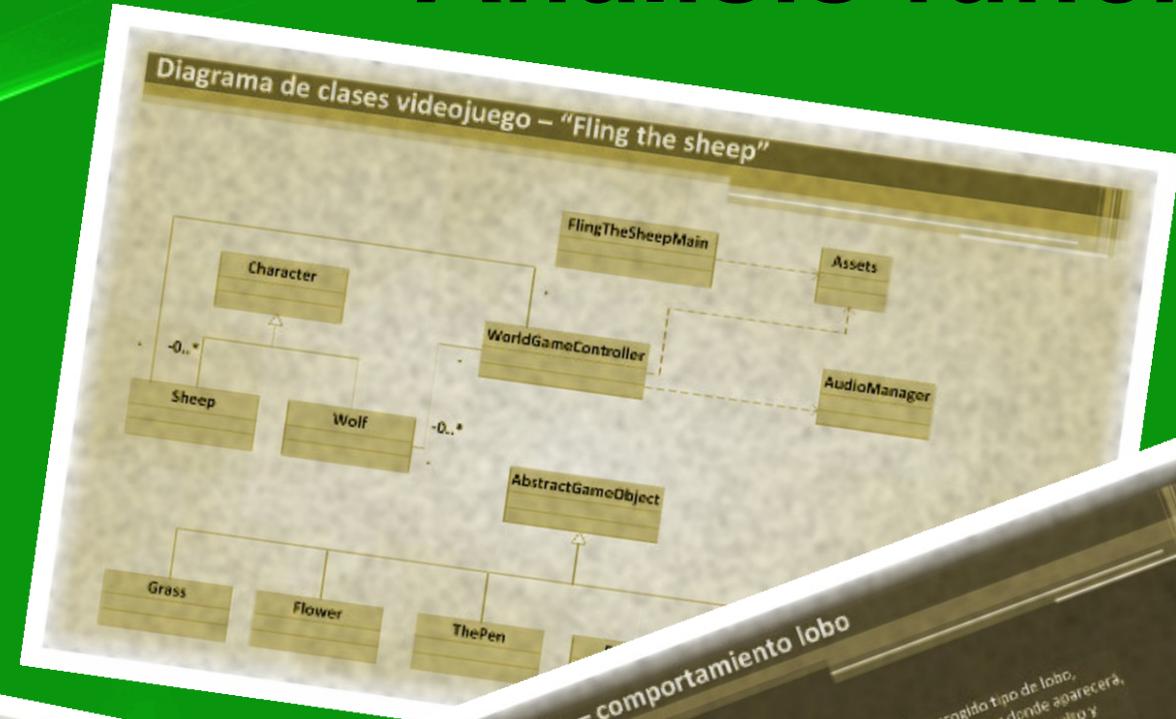
Finalmente, existe el riesgo de que el gameplay definido en el diseño conceptual del videojuego, una vez puesto en práctica, no sea del todo funcional o necesite de un balanceo para conseguir un juego equilibrado y jugable



# FLING THE SHEEP

# Análisis funcional

- Casos de uso
- Diagramas de clases
- Máquinas de estados

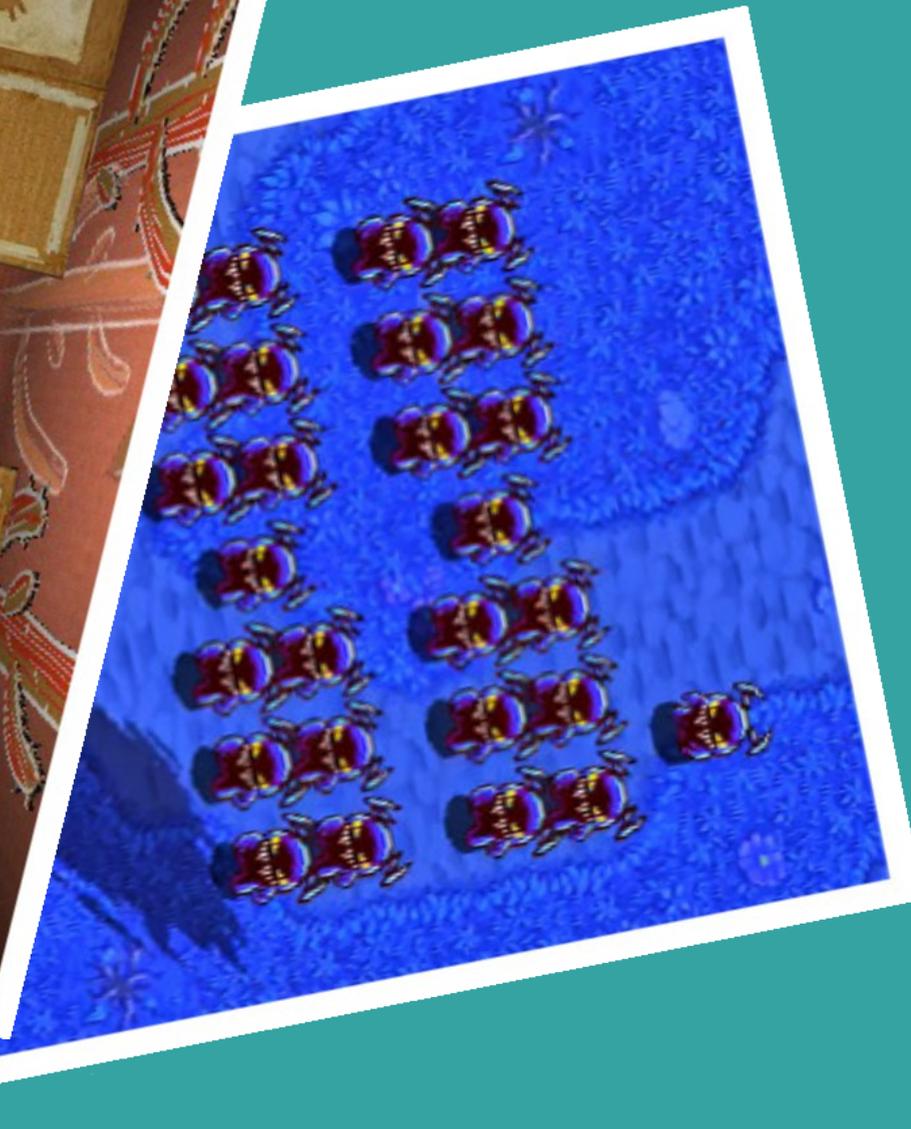
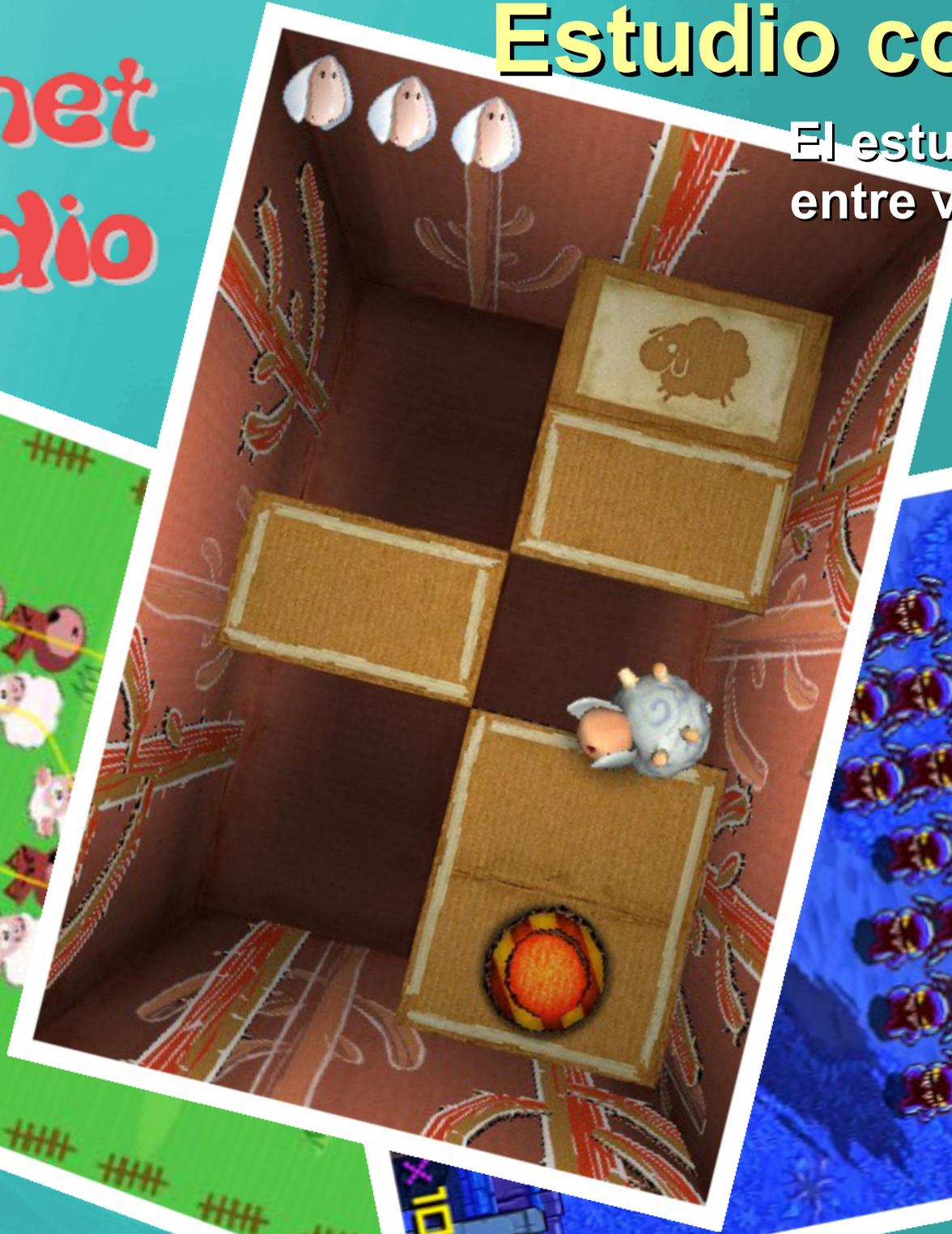


**F L I N G**  
**T H E**  
**S H E E P**

# 3 Comet Studio

## Estudio comparativo

El estudio se ha realizado entre varios juegos, entre ellos:



- Sheep & Wolf
- Sheep Up !
- Attack of the Spooklings

# Perfiles de usuarios

# 3 Comet Studio

F L I N G  
T H E  
S H E E P

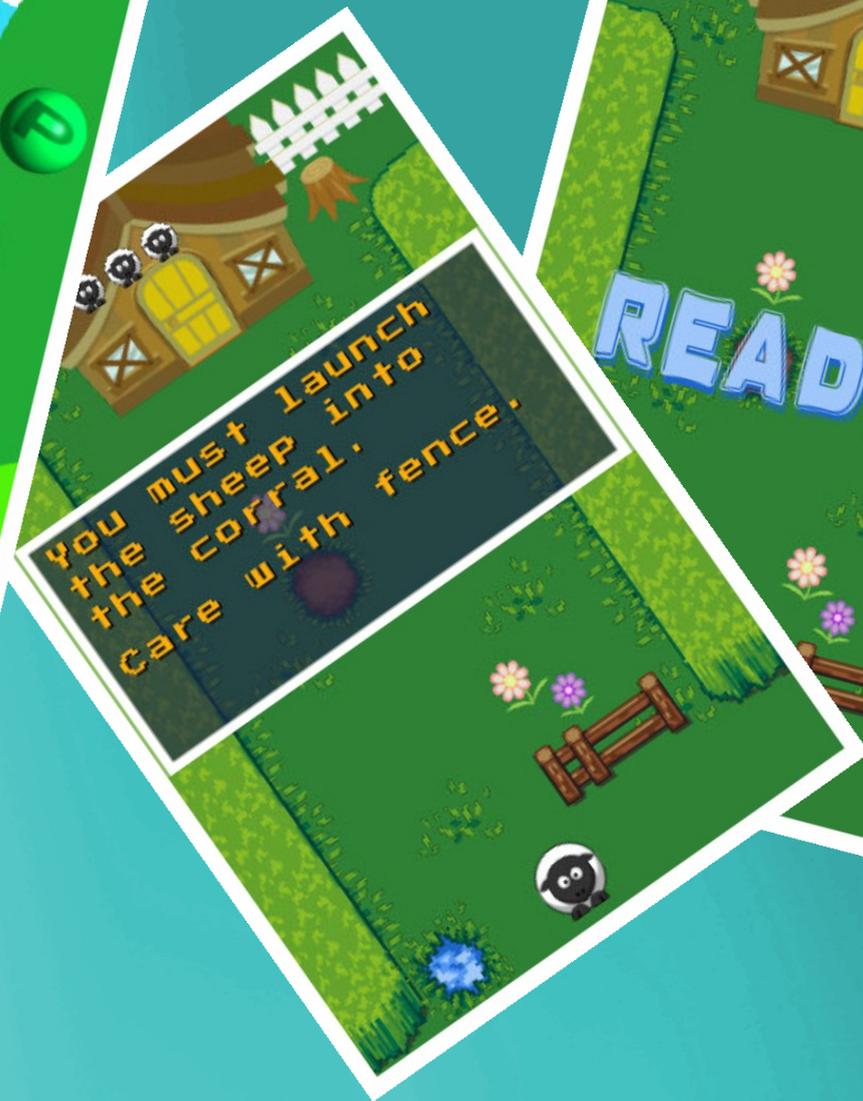
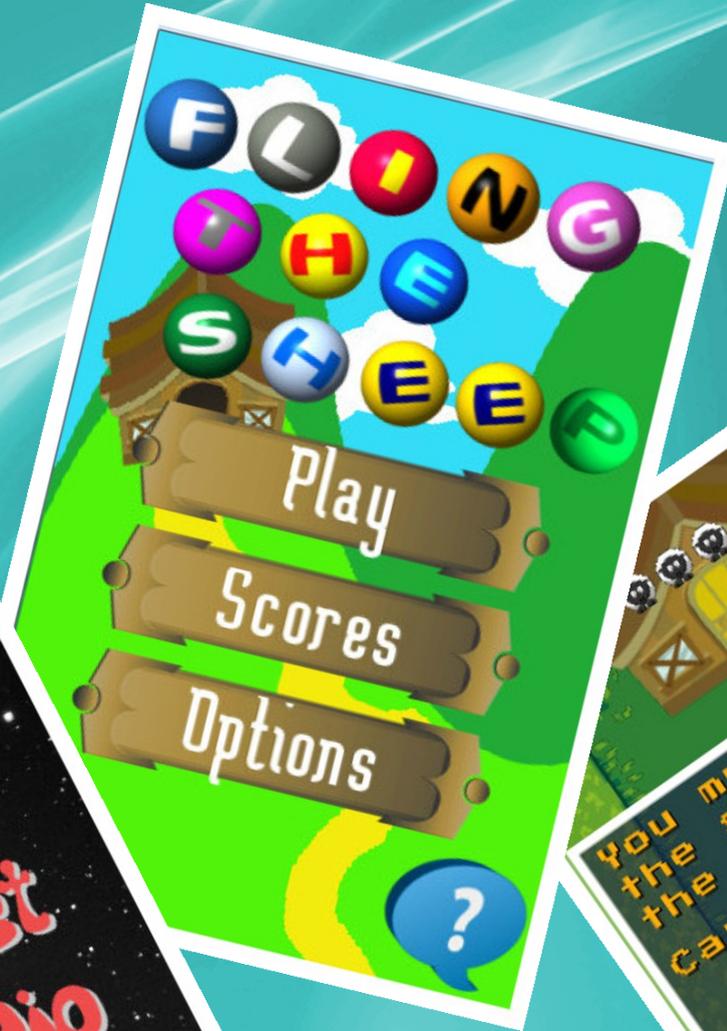
Los juegos casuales, son juegos con mecánicas simples. Este hecho hace que estén al alcance de cualquier usuario aunque éste no sea un jugador experimentado acostumbrado a utilizar juegos con mecánicas más complicadas que requieren de cierta experiencia previa. Al mismo tiempo, son juegos que pueden ser jugados en cortos espacios de tiempo. Por tanto, el tipo de usuario de estos juegos será muy variado.

Gracias a los smarphones y la sencillez de las mecánicas de juego provoca que estos juegos esten al alcance de grandes masas de usuarios.



# Diseño "Concept Art"

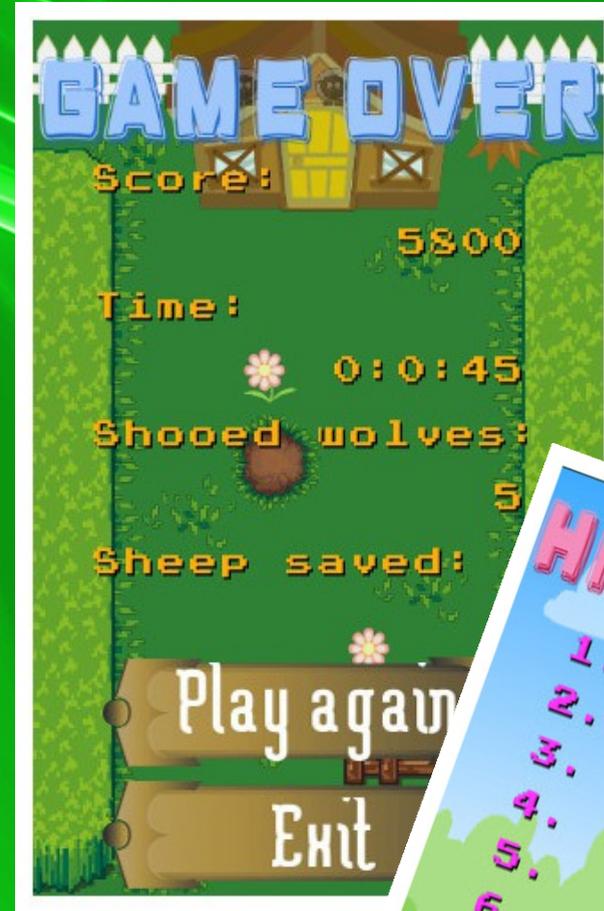
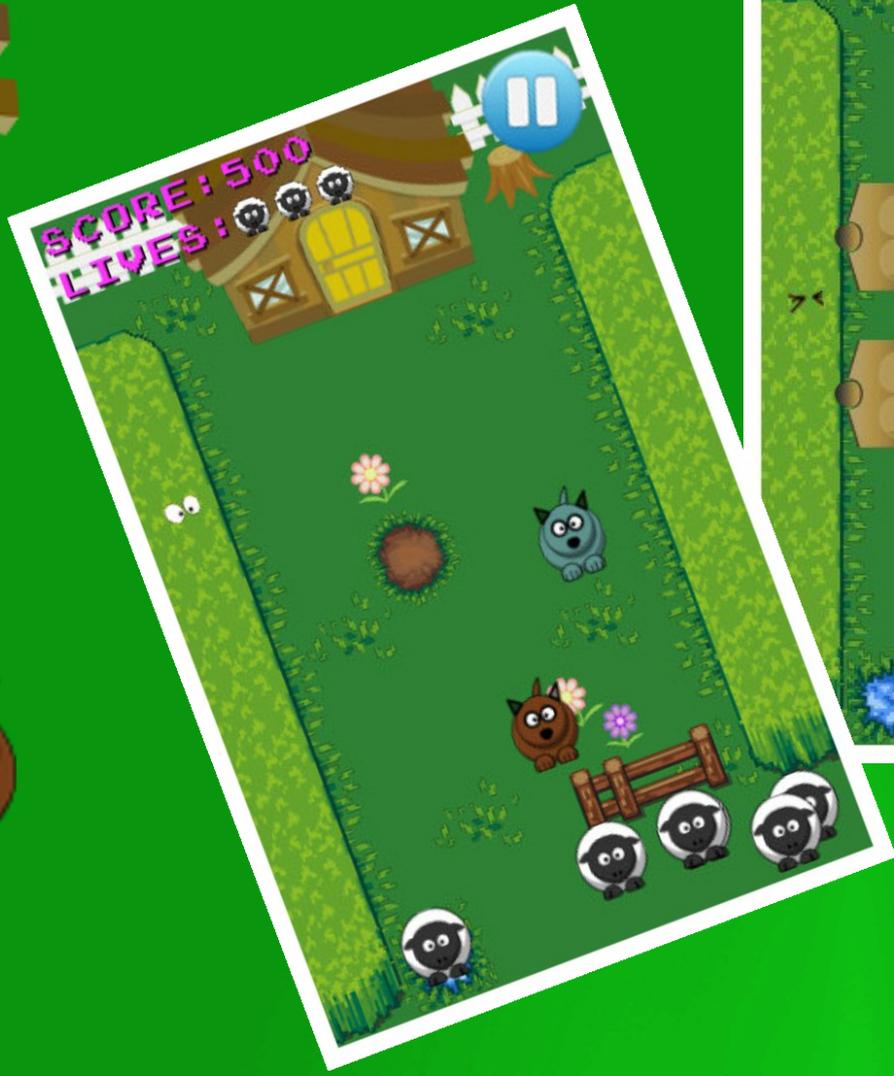
# 3 Comet Studio



FLING THE SHEEP

# 3 Comet Studio

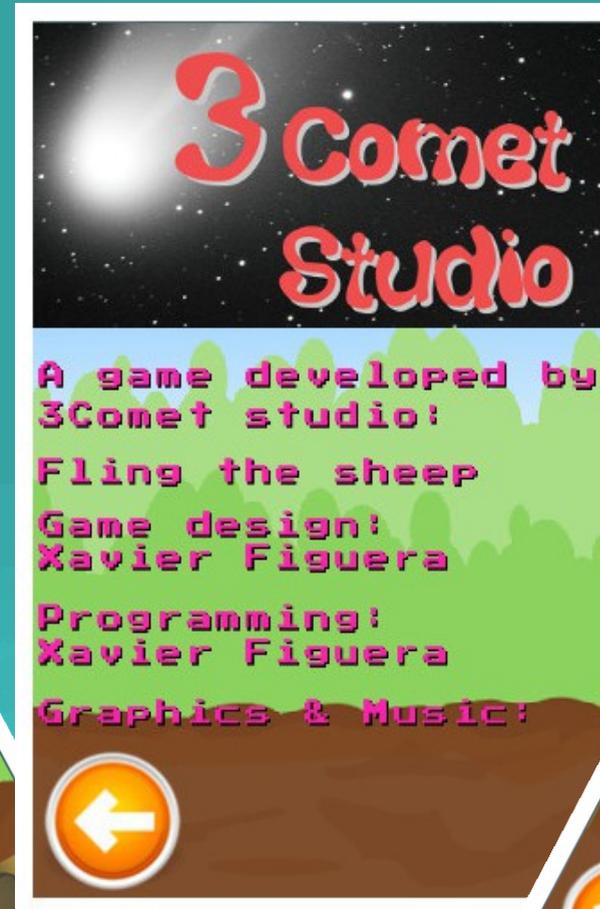
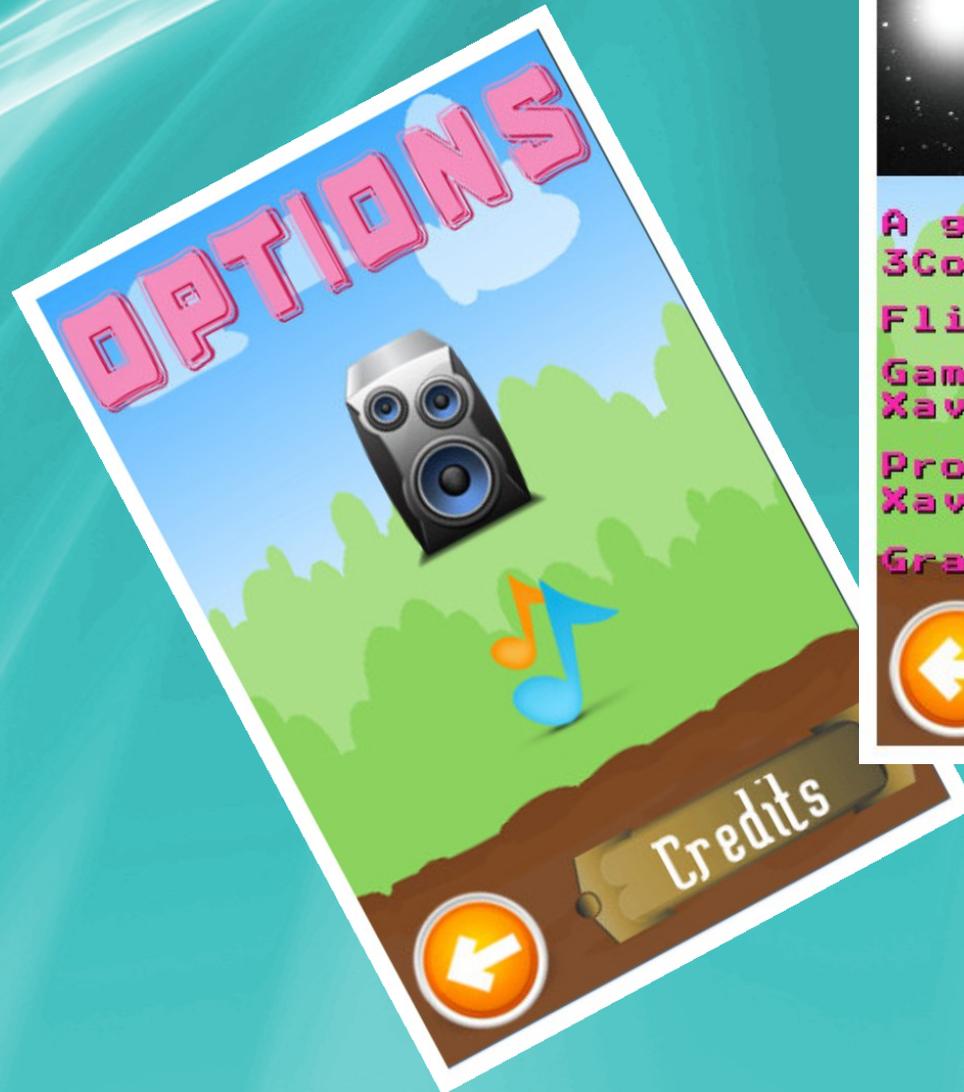
## Diseño "Concept Art"



FLING THE SHEEP

# 3 Comet Studio

## Diseño "Concept Art"



# Implementación

## 3 Comet Studio

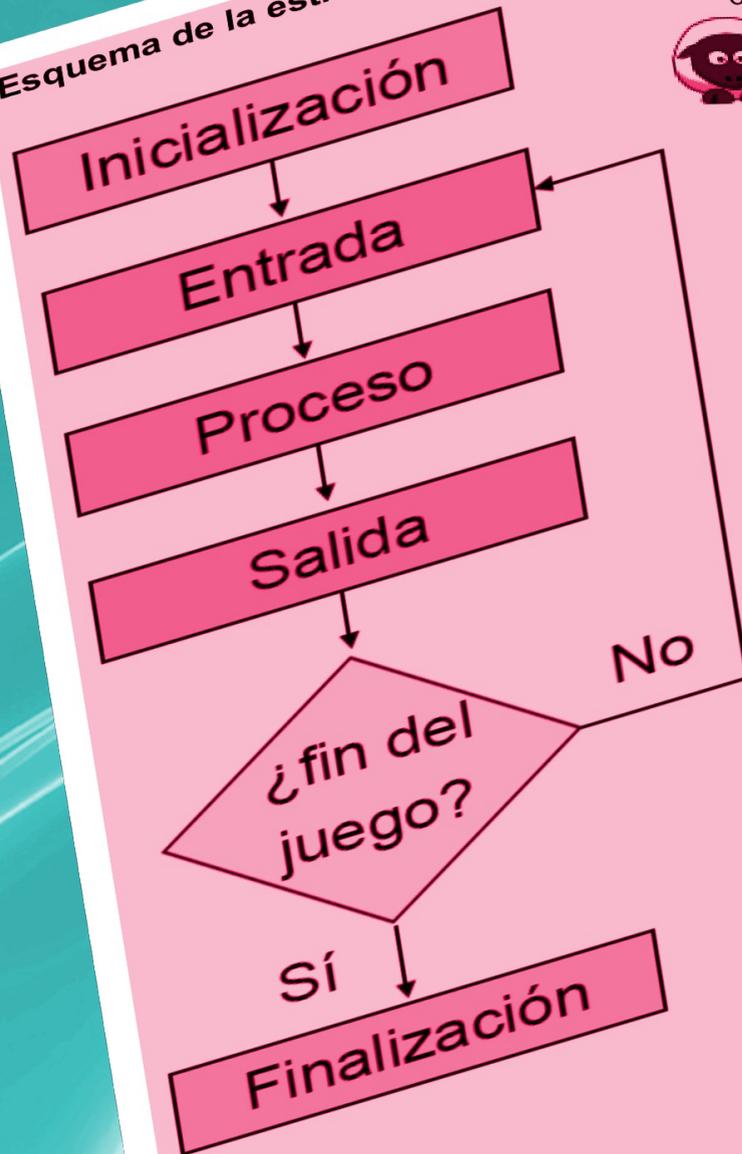


LibGDX es un framework multiplataforma que actualmente soporta Windows, Linux, Mac OS X, Android, iOS y HTML5. Esto permite escribir el código una sola vez y desplegarlo en las distintas plataformas comentadas, sin tener que modificar el código. LibGDX permite bajar a más bajo nivel si se desea, ya que da acceso directo al sistema de ficheros, dispositivos de entrada, de sonido y a OpenGL a través de la interfaz unificada OpenGL ES1.1 y 2.0.

Una aplicación convencional sigue una secuencia lógica lineal, reacciona cuando sucede algún evento, es decir responde a eventos como por ejemplo cuando se pulsa una tecla. Por el contrario un videojuego es más complejo, hace cálculos y dibuja en pantalla constantemente, aunque no se produzca ningún evento.



## Esquema de la estructura básica de un videojuego



**Inicialización:** Creación de estructuras de datos, carga de recursos gráficos y sonidos



**Entrada:** Gestión de pantalla táctil, joystick, teclado.



**Proceso:** Lanzamos una oveja al corral, rebota la oveja, salta un lobo al escenario etc.



**Salida:** Mostrar gráficos y reproducir sonidos.



**Finalización:** Liberar recursos.

# 3 Comet Studio



Programar videojuegos es una forma completamente diferente de programar software. La diferencia principal entre una aplicación convencional y un videojuego radica en la gestión de eventos: un programa ofrece respuestas a eventos y, una vez son servidos, permanece a la espera de una nueva orden, en cambio, un videojuego es un programa que debe actuar en tiempo real, tiene que estar haciendo cálculos y dibujando en pantalla constantemente. No se puede esperar a que suceda un evento para poder actuar: aunque el jugador no haga nada, no presione una sola tecla, no mueva el ratón, el juego tiene que calcular el tiempo que lleva jugando en ese nivel, si le va a atacar algún enemigo, si está cargando energía y, por supuesto, dibujar en pantalla todo lo que va sucediendo.

La mayoría de videojuegos están construidos bajo la misma estructura básica sobre la cual corre el programa. Esta estructura puede tener variaciones, pero de forma general la estructura queda recogida en la fotografía de la izquierda.

## Estructura de un videojuego



FLING THE SHEEP

# 3 Comet Studio

# Entorno de desarrollo

LibGDX permite desarrollar, para varias plataformas sin casi modificar nada en el código, esto hecho permite ejecutar el desarrollo en el escritorio, sin tener que ejecutarlo forzosamente en un dispositivo real o virtual. Esto provoca que el desarrollo sea mucho más fluido en líneas generales.



The screenshot displays the Eclipse IDE interface for the 'FlingTheSheep' project. The Package Explorer on the left shows the project structure, including packages like 'com.threecomet.flingthesheep' and 'com.threecomet.flingthesheep.game'. The main editor shows the 'WorldGameController.java' file with the following code:

```
if(playingMusic != null) playingMusic.play();
}

public void PlayMusic(Music music,float volume,boolean loop)
{
    if(!GamePreferences.instance.music) return;
    playingMusic = music;
    music.setLooping(loop);
    music.setVolume(volume);
    music.play();
}

public void StopMusic()
{
    if(playingMusic != null) playingMusic.stop();
}
```

Below the code, a UML Class Diagram shows the relationships between classes: 'Object' (java.lang), 'Assets' (com.threecomet.flingthesheep.game), and 'WorldGameController' (com.threecomet.flingthesheep.game). 'WorldGameController' has a dependency on 'Assets' and an instance of 'Assets' (indicated by a dashed arrow with '+instance' and '0..1').

On the right, a preview window shows the game running on a mobile device. The game screen displays a score of 2700, lives, and a sheep character. The bottom right corner of the IDE shows a list of classes and a small diagram.



# 3 Comet Studio

## Patrones de diseño



En el diseño de cualquier aplicación como por ejemplo un videojuego, no es posible abordarlo por completo. El proceso de diseño de una aplicación suele ser iterativo y se realiza en diferentes etapas de forma que se vaya refinando con el tiempo. Conseguir un diseño perfecto a la primera es muy difícil de conseguir. Es en el diseño donde se definen las estructuras y entidades que se van a encargar de resolver el problema planteado. Como de bien se definan estas estructuras y entidades influirá en gran medida, en el éxito o fracaso del proyecto y en la viabilidad de su mantenimiento.



Los patrones de diseño son formas reconocidas y probadas maneras de resolver problemas de diseño que son recurrentes en el tiempo. No es más que basarse en experiencias anteriores en problemas similares aplicando soluciones probadas anteriormente en determinados contextos para alcanzar un mejor diseño más rápidamente.

Existen distintos patrones, para distintos problemas de diseño. No obstante, para este proyecto se ha utilizado el patrón singleton, ya que era necesario tener una única instancia de algunas determinadas clases.



```
package com.threecomet.flingthesheep.game;

import com.badlogic.gdx.Gdx;

public class CounterTime
{
    private static final String LABEL = CounterTime.class.getName();

    public static final CounterTime instance = new CounterTime();

    private boolean isStarted;
    private boolean isPaused;
    private long timeStartGame;
    private long totalTimeGame;
    private long totalTimePausedGame;
    private long collectorTimePausedGame;
    private float time;
    private int seconds;

    //singleton
    private CounterTime() {}

    public void init()
    {
```

Singleton
-instance: Singleton
+instance(): static Singleton
-Singleton()

# Conclusiones

3 Comet  
Studio



El desarrollo del proyecto ha sido tal y como se esperaba en líneas generales, no han habido grandes desviaciones de tiempo ni durante el diseño ni en el posterior desarrollo de este.

El hecho de haber escogido libGDX para el desarrollo ha sido una buena elección. Es también una buena elección escoger un framework o un motor focalizado al desarrollo de videojuegos, si se quiere implementar uno ya sea para Android o para cualquier plataforma.

Existe la posibilidad de desarrollarlo sin ningún tipo de framework o motor, pero hay que tener claro que se quiere hacer exactamente y a donde se quiere llegar, es decir si lo que se quiere es centrarse en la programación de videojuegos, es mucho mejor partir de un framework o motor ya que de esta forma nos desligamos de programar ciertos apartados que están más focalizados a la programación de librerías o motores para videojuegos que no a la programación de videojuegos en sí.

Yendo más allá de donde el programador se quiere centrar, existe otro factor importante a tener en cuenta en el momento de plantear un proyecto llamado tiempo, con el tiempo de que se ha dispuesto para este proyecto, queda aún más justificado la elección de este framework para realizarlo.



# Lineas abiertas

3 Comet  
Studio



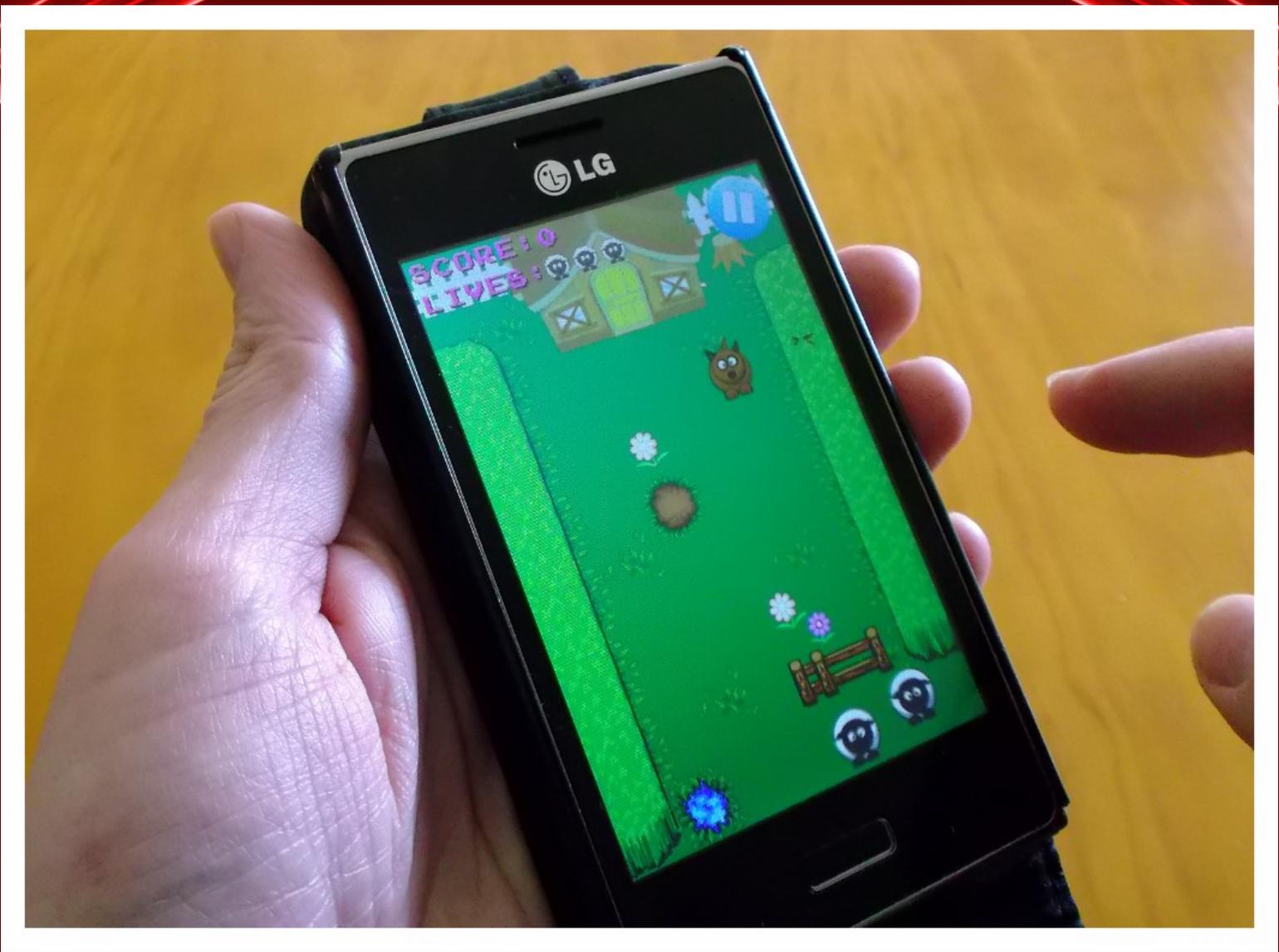
- Se hará un repaso al proyecto por si existe algún bug para solucionarlo.
- Adaptación a distintas resoluciones de pantalla.
- Añadir los power-ups que se definieron al principio del proyecto, por temas de tiempo han quedado fuera de la entrega.
- Balancear el gameplay, añadiendo alguna mecánica de juego extra.
- Publicar el juego en Google Play.
- Crean un blog del juego, se mirara la posibilidad de hacer funcionar el juego en el blog con WebGL. <http://projects.3comet.com/>
- Ampliar la tabla de puntuaciones:
  - La tabla no solamente guardara los puntos, también guardara el tiempo jugado, número de lobos ahuyentados, número de ovejas salvadas y nombre del usuario con 5 caracteres.
  - Crear un registro de puntuaciones global, consultable desde el blog del videojuego.
- Crear más niveles de juego, con distribuciones diferentes, algún nivel podría también tener mecánicas diferentes de juego con los mismos personajes.
- .....



FLING  
THE  
SHEEP

# Videojuego en un LG-610v

3 Comet  
Studio





**© 2013 Xavier Figuera Alberich**  
**3Comet studio**

