

Proyecto Final de Carrera: Videojuegos educativos

Kamikawa, videojuego para practicar el vocabulario de la lengua inglesa.

Antonio Javier Garrido Rodríguez
2º ciclo de Ingeniería en Informática
antoniojaviergr@gmail.com

Heliodoro Tejedor Navarro

1. Justificación
2. Objetivos PFC
3. Enfoque y método seguido
4. Recursos necesarios
5. Planificación
6. Requisitos
7. Mecánica del juego
8. Arquitectura
9. Implementación
10. Resultado
11. Pruebas
12. Conclusiones

1. Justificación

En la elección de mi PFC apliqué como principal criterio que fuese útil y provechoso.

Teniendo en cuenta que los teléfonos inteligentes están proliferando a un ritmo exponencial, y más del 80% de éstos funcionan con el SO de Android. Creo que es muy interesante el desarrollo de aplicaciones para dicho SO. Porque estaremos desarrollando aplicaciones con la posibilidad de estar presente en millones de dispositivos.

Por lo tanto, me parece una gran oportunidad y buena idea, aprender a desarrollar aplicaciones para dispositivos con Android; y aprovechar la realización de este trabajo para iniciarme en ello. Además nunca había implementado un videojuego de cierta envergadura, sólo juegos muy sencillos, y con fines puramente didácticos.

2. Objetivos PFC

Objetivos personales: el principal es aprender sobre el diseño y desarrollo de aplicaciones para Android. Este lleva implícito a su vez los siguientes objetivos:

- Conocer cómo funciona el SO Android.
- Aprender a utilizar el Framework de Android para el desarrollo de aplicaciones.
- Conocer y utilizar parte de las funcionalidades ofrecida en sus APIs.
- Tener alguna experiencia en el desarrollo para Android.
- Aprender algo sobre el diseño y desarrollo de videojuegos.

En cuanto a los objetivos del videojuego: debemos ser conscientes y tener en cuenta el contexto al que se estará expuesto cuando se juega. Es decir, la mayoría de jugadores de videojuegos para dispositivos móviles juegan en sus “tiempos muertos”. En consecuencia, los objetivos a tener en cuenta deben ser:

2. Objetivos PFC

- Que las partidas sean cortas.
- Fácil de jugar.
- Con una mecánica sencilla.
- Pausable. Ya que estará expuesto a continuas interrupciones.

Otro objetivo propuesto es desarrollar el videojuego completamente desde cero, es decir, no utilizar ningún motor de desarrollo para videojuegos. Toda la funcionalidad será implementada por mí.

3. Enfoque y método seguido

El enfoque inicial fue el siguiente:

1. Tener claro o lo suficientemente claro **qué** debo hacer. Para ello, debía leer e informarme sobre videojuegos para dispositivos móviles.
2. Aprender **cómo** debo hacerlo. Preparar el entorno de desarrollo; aprender los conocimientos básicos sobre la plataforma de desarrollo de Android y realizar algunas aplicaciones básicas y sencillas; aprender lo básico sobre el desarrollo de videojuegos.
3. Una vez sabido qué debo hacer y cómo, estaba en condiciones de poder hacer estimaciones. Otro factor importante y determinante es el tiempo, así que teniendo en cuenta los factores de: **qué** hacer, **cómo** hacerlo y el **tiempo** disponible; estaba en disposición de poder hacer **estimaciones** sobre la magnitud del resultado que podría obtener.

3. Enfoque y método seguido

Dado que no tenía ninguna experiencia en el diseño de videojuegos, tomé como referencia e inspiración uno existente, Letris; basándome en él, diseñé el mío.

Método seguido: principio de “**desarrollo ágil de software**”. Donde rápidamente obtengo una demo (versión inicial ejecutable y con algún tipo de resultado), y mediante iteraciones, voy añadiendo funcionalidades hasta obtener el resultado final.

4. Recursos necesarios

Nombraré los recursos utilizados en mi caso personal, todos son sustituibles por otros que cumplan sus mismas funciones, excepto los propios de la plataforma de desarrollo de Android.

Recursos hardware:

Ordenador de sobremesa: AMD Intel Core Duo a 3 GHz y 4 GB de memoria RAM.

Dispositivos móviles auxiliares:

- Sony Xperia S: utilizado durante toda la fase de desarrollo para probar el código implementado.
- Para las pruebas finales de adaptabilidad: Smartphone LG Optimus L7, Smartphone Samsung Galaxy S3 y Tablet Galaxy Tab 3 10.1

4. Recursos necesarios

Recursos software:

Eclipse : IDE utilizado para la programación del videojuego.

JDK : El sistema operativo de Android se basa en el lenguaje de programación Java. Por tanto, necesitamos el intérprete de Java.

SDK Android : necesario para programar aplicaciones para el SO de Android.

Plugin ADT para Eclipse: Una vez instalado el SDK, necesitamos incluir y configurar el ADT en Eclipse, éste permite la integración entre el SDK y el IDE Eclipse.

6. Requisitos

Requisitos funcionales

Teniendo en cuenta el contexto de los videojuegos para dispositivos móviles:

- Mecánica de juego sencilla.
- Intuitivo y fácil de jugar.
- Partidas cortas.
- Pausable.
- Que incite a la competencia y superación de objetivos. Para ello debe persistir datos en el dispositivo.

6. Requisitos

Requisitos no funcionales

En principio el videojuego es compatible con todas las versiones de Android desde la 3.2 en adelante, y con todo tipo y tamaños de dispositivos; ofreciendo un óptimo resultado de ejecución en la versión más popular, Jelly bean.

Los requisitos no funcionales que debe satisfacer son los siguientes:

- Compatibilidad con el mayor número de versiones de Android.
- Compatibilidad con el mayor número de diferentes tipos y tamaños de dispositivos.
- El idioma será español.
- Disponer de un Log con información sobre la ejecución.
- No utilizar ningún motor de desarrollo.

7. Mecánica del juego

El juego consiste básicamente en formar palabras en la lengua inglesa. Para ello disponemos de un escenario, en el cual, van apareciendo letras del abecedario. Éstas caminan por la pantalla en las cuatro direcciones posibles, izquierda, derecha, arriba y abajo. Caminan de forma aleatoria hasta llegar a un límite de pantalla o colisionar con otro agente letra, cambiando su dirección. Las letras van apareciendo individualmente, de forma progresiva y a un ritmo constante. La frecuencia de aparición y la velocidad de movimiento, será determinado por el nivel en el que se encuentre el videojuego, a mayor nivel, mayor velocidad de movimiento y frecuencia de aparición.

El modo de formar una palabra es haciendo pulsaciones con el dedo sobre cada una de las letras que la representen. El juego cuenta con un *display* el cual visualiza la palabra que se está formando en cada momento. Si la palabra actual es reconocida como palabra válida en la lengua inglesa, el *display* cambiar al color verde, sino continua en rojo.

7. Mecánica del juego

La dificultad en cada nivel vendrá determinada por la frecuencia de aparición y la velocidad de movimiento de los agentes letras, en el primer nivel estos parámetros tendrán los valores mínimos posibles e irán aumentando progresivamente a medida que avancen los niveles. El videojuego cuenta con 10 niveles, una vez superado llegaremos al final del juego.

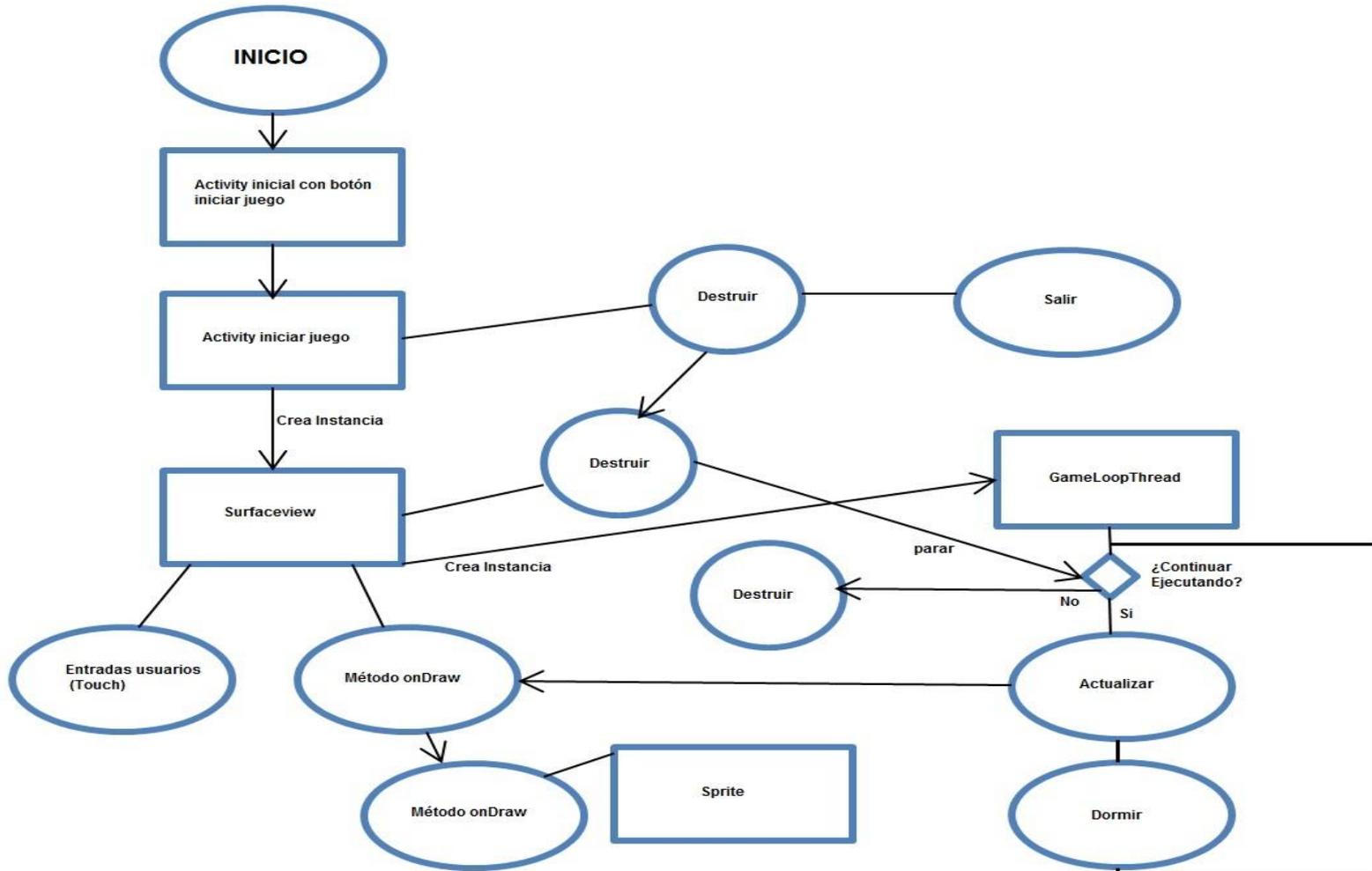
La puntuación se consigue a través de las palabras que se vayan formando. El número de puntos asociados a cada palabra dependerá de la longitud de éstas, a mayor longitud de palabra mayor puntuación.

7. Mecánica del juego

Para finalizar su formación y confirmación de la misma, hay que hacer una pulsación sobre el *display* cuando la palabra sea reconocida como válida. Con ello, eliminamos de la pantalla a los agentes letras que formen la palabra, disminuyendo el número de estos en pantalla y recibiendo la puntuación que proceda en función a la longitud de la palabra. Si se desea, es posible desmarcar letras de una palabra inválida, mediante una pulsación sobre el *display*.

Para superar un nivel hay que permanecer el tiempo de duración del mismo, formando palabras sin exceder en ningún momento el número máximo de agentes letras permitidos en la pantalla. El tiempo de duración de cada nivel es de 2 minutos, y el número máximo de agentes letras permitidos simultáneamente en la pantalla es de 25, para los niveles desde el 1 hasta el 5, ambos inclusive; y de 50 para el resto.

8. Arquitectura



9. Implementación

Ha sido la fase más costosa, en cuanto a tiempo y dedicación.

Una vez implementada la arquitectura básica funcional, fui añadiendo funcionalidades hasta obtener el resultado final.

Para consultas técnicas sobre Android y su programación he consultado sobre todo la guía para desarrolladores de Android, fuente inestimable de información. Otra fuente consultada con frecuencia ha sido la web “Stack Overflow”, muy útil.

10. Resultado

La versión final del videojuego implementa todas las funcionalidades definidas como básicas en el documento de diseño. Las funcionalidades establecidas como “opcionales” (si el tiempo disponible lo permitía), quedan como posibles mejoras en las *Líneas de trabajo pendiente*.

A continuación muestro un video donde ejecuto y utilizo el videojuego.

11. Pruebas

El videojuego en principio, debe funcionar correctamente en cualquier tipo de dispositivo. Para comprobarlo, he realizado pruebas en estos dispositivos, que a continuación veréis en los videos:

- Smartphone Sony Xperia S con sistema operativo Android versión 4.1.2 y una pantalla de 720 x 1280 pixeles y densidad extra alta. La visualización del video anterior lo demuestra.
- Smartphone LG Optimus L7 con sistema operativo Android versión 4.0.3 y una pantalla de 480 x 800 pixeles y densidad media.
- Smartphone Samsung Galaxy S3 con sistema operativo Android versión 4.1.2 y una pantalla de 720 x 1280 pixeles y densidad extra alta.
- Tablet Galaxy Tab 3 10.1 con sistema operativo Android versión 4.2.1 y una pantalla de 1280 x 800 pixeles y densidad media.

11. Pruebas

Para comprobar que se adapta a los diferentes tipos de tamaños y densidades de pantalla, hice algunas pruebas específicas. Comprobando que utiliza los recursos de representación gráfica (como layout y drawable) en función a las características del dispositivo.

Una de las pruebas consiste en comprobar que utiliza la carpeta *drawable* correcta en función a la densidad de pantalla del dispositivo. Para ello, diferencio las imágenes de cada carpeta, y lo hago coloreando de un color diferente el botón *acelerar* (y la letra A para alta densidad). Utilizando los siguientes colores:

Baja densidad: color rojo.

Media densidad: color naranja.

Alta densidad: color amarillo.

Extra alta densidad: color violeta.

En el vídeo podremos comprobar cómo el videojuego utiliza unas imágenes u otras en función a la densidad de pantalla.

12. Conclusiones

He aprendido de forma práctica -hasta ahora sólo era consciente a nivel teórico- que para la realización de un proyecto de considerable envergadura, un factor importantísimo para el éxito y obtener el producto para la fecha prevista, es realizar una buena planificación e ir cumpliendo con todos los hitos en el tiempo estimado.

He cumplido uno de los principales objetivos personales: “aprender y adquirir práctica en el diseño y desarrollo de aplicaciones para Android”.

En cuanto a los objetivos propios del juego, también creo que he cumplido con todos los marcados. El videojuego se adapta perfectamente al contexto de videojuego para dispositivos móviles.

En general, estoy satisfecho con el resultado conseguido.